# SAM 68K

# Handbuch

(C) Copyright 1985 by
kws Computersysteme GmbH

D 7505 Ettlingen

W.Germany

Die in diesem Handbuch zusammengestellten Informationen wurden sorgfältig geprüft und sollten fehlerfrei sein. Dennoch übernimmt KWS keine Verantwortung für Schäden oder Schwierigkeiten, die sich daraus aufgrund von Ungenauigkeiten oder Irrtümern bei der Anwendung oder dem Gebrauch der Produkte ergeben. KWS behält sich das Recht vor, Änderungen an Hard- und Software durchzuführen, die die Zuverlässigkeit und Funktionsweise verbessern oder die Bedienung erleichtern. Diese Änderungen müssen nicht in jedem Einzelfall dokumentiert werden. Alle Produkte von KWS unterliegen dem Urheberrecht oder sind gesetzlich geschützt. Sie dürfen nicht ohne Lizenzvertrag kopiert oder nachgebaut werden.

# Inhaltsverzeichnis


1 Einleitung

2 Allgemeine Beschreibung

2.1 Kurzbeschreibung
2.2 Technische Daten SAM 68K
2.3 Lieferumfang
2.4 Systemkonfiguration

3 Inbetriebnahme

3.1 Einführung
3.2 Sichtprüfung
3.3 Vorbereitung des Systems
3.4 Anschluß der Floppy-Disk-Laufwerke
3.5 Anschluß eines Druckers
3.6 Datenformat der V.24-Schnittstellen

4 Monitorprogramm von SAM 68K

4.1 Einführung
4.2 Versions- und Revisionsnummer
4.3 Format der Kommandozeile
4.4 Register anzeigen und ändern
4.5 Speicherzellen anzeigen und ändern
4.6 Starten und Testen eines Programms
4.7 Ein/Ausgabekanäle
4.8 Laden und Abspeichern von Programmen
4.9 Starten des Betriebssystems
4.10 Abbrechen eines Programms
4.11 Monitorroutinen für den Benutzer
4.12 Video- und Tastaturtreiber

5 Kopplung mit Hostsystemen

5.1 Einführung
5.2 Praktische Durchführung

6 Hardwarebeschreibung

6.1 Einführung
6.2 Reset- und Break-Schaltung
6.3 Speicher
6.4 Adressdekodierung und Steuersignale
6.5 Speicherbelegung
6.6 Ein/Ausgabe-Bausteine
6.7 Interrupts
6.8 Bus Interface
6.9 Timeout
6.10 Floppy-Disk-Controller

6.11 Video-Controller

7 Tabellen und Schaltpläne

7.1 Speicherbelegung und Adressen der Ein/Ausgabe-Bausteine
7.2 Belegung des Steckverbinders PF1 für serielle Ein/Ausgabe
7.3 Belegung des Steckverbinders PF2 für allgemeine Ein/Ausgabe
7.4 Belegung der Steckverbinder PF3 und PF4 für Floppy-Disk-Anschluß
7.5 Belegung des Bussteckers VG 1
7.6 Belegung des Bussteckers VG 2
7.7 Einstellung der Baud-Rate der V.24 Schnittstellen
7.8 Lageplan der Jumper
7.9 Bedeutung der Jumper
7.10 Schaltplan des Ein/Ausgabeteils
7.11 Schaltplan des Floppy-Disk-Anschlusses
7.12 Schaltplan des Businterfaces
7.13 Verwendetes Format für Binärcode
7.14 RAM - Belegung durch Software
7.15 Steuerzeichen der Videotreiber
7.16 Belegte Trap-Vektoren und Interrupt-Ebenen
7.17 Belegung des Steckverbinders PF5 für den Anschluß eines Videomonitors
7.18 Belegung des Steckverbinders PF6 für RESET NMI und HALT-LED

8 Sondertypen SAM-68K mit 10 MHz Prozessortakt und erhöhter Grafikauflösung

8.1 Einleitung
8.2 Auswirkungen des 10 MHz Prozessortaktes
8.3 Organisation des Video-RAMs bei erhöhter Videoauflösung
8.3 Monitorprogramm für die Sondertypen

9 Kabelplatine (Rueckwandverdrahtung RTC 1)

9.1 Schaltplan der RTC 1
9.2 Anmerkungen zum Schaltplan

# 1 Einleitung

Das vorliegende Handbuch ist die technische Dokumentation zu Hard- und Firmware der Rechnerplatine SAM 68K. Es enthält alle Informationen die nötig sind, um SAM 68K als Einzelplatine in Betrieb zu nehmen und seine Funktionsweise zu verstehen. Außerdem enthält das Handbuch eine ausführliche Beschreibung des Monitorprogramms von SAM 68K. Im Abschnitt "Tabellen und Schaltpläne" wird u.a. die Adreßzuordnung von RAM, EPROM und Peripheriebausteinen sowie die Belegung der Ein/Ausgabe- und Bussteckverbinder beschrieben.

Die Platine SAM 68K gibt es in verschiedenen Versionen. Das vorliegende Handbuch ist NUR FUER HARDWARE-VERSIONEN AB V7 gültig.

Ab der Version V7 gibt es die SAM 68K Platine in den folgenden Konfigurationen, die sich hinsichtlich der Größe des (auf der Platine befindlichen) RAMs, der Taktfrequenz des Prozessors und der Auflösung des Videoteils unterscheiden:

| Typ | Bezeichnung | RAM | Takt | Videoauflösung (max.) |
|---|---|---|---|---|
| 1 | SAM-68K/256-08-N | 256 KB | 8 MHz | 512x512 Punkte |
| 2 | SAM-68K/640-08-N | 640 KB | 8 MHz | 512x512 Punkte |
| 3 | SAM-68K/256-10-H | 256 KB | 10 MHz | 768x480 Punkte |
| 4 | SAM-68K/640-10-H | 640 KB | 10 MHz | 768x480 Punkte |
| 5 | SAM-68K/256-08-H | 256 KB | 8 MHz | 768x480 Punkte |
| 6 | SAM-68K/640-08-H | 640 KB | 8 MHz | 768x480 Punkte |

Die Typen 1 und 2 sind dabei als die Standardtypen anzusehen, die von kws optimal unterstützt werden. Der Einsatz der Typen 3 und 4, sowie insbesondere der Typen 5 und 6 sollte in jedem Fall nur nach vorheriger Rücksprache mit kws erfolgen.

WICHTIG:

Das vorliegende Handbuch beschreibt in den folgenden Kapiteln die Typen 1 und 2. Davon abweichende Daten der restlichen Typen sind in einem Anhang A dokumentiert.

Wenn Sie SAM 68K als Einzelplatine in Betrieb nehmen wollen, so folgen Sie insbesondere den Ausführungen im Kapitel 3 "Inbetriebnahme", das Ihnen Schritt für Schritt erklärt, was Sie tun müssen, damit sich SAM 68K zum ersten Mal meldet.

# 2 Allgemeine Beschreibung

## 2.1 Kurzbeschreibung

SAM 68K ist ein vollständiger Einplatinencomputer auf der Basis des MC68000, dem zur Zeit leistungsfähigsten 16/32-Bit Prozessor. Er enthält auf einer einzigen Platine alle die Baugruppen, die man benötigt, um ein Mikroprozessorsystem der technologischen Spitzenklasse aufzubauen.

Im einzelnen bietet SAM 68K:

* 16/32 Bit Hochleistungsprozessor MC 68000
* 256 KB oder 640 KB dyn. RAM
* bis zu 64 KB EPROM
* Floppy-Disk-Controller für 5"- und 8"-Laufwerke
* Leistungsfähiger Videoteil für Schrift und hochauflösende Grafik (bis zu 512 x 512 Pixel)
* zwei V.24 Schnittstellen (serielle E/A), konfigurierbar auch als schnelle synchrone Datenleitungen mit Übertragungsraten bis 800KB
* Anschluß für Tastatur und Videomonitor
* 20 parallele Ein/Ausgabeleitungen
* zwei Timer, Schieberegister
* voll gepuffertes Businterface, kompatibel zu allen Baugruppen der EB 68000 Reihe
* leistungsfähiges Monitorprogramm
* Softwareoptionen auf Wunsch z.B. verschiedene Betriebssyteme mit höheren Programmiersprachen, Editoren, Assembler, Grafiksoftware u.a.

Aufgrund seiner vielseitigen Erweiterungsmöglichkeiten und der großen Leistungsfähigkeit des Prozessors eignet sich SAM 68K besonders für schnelle Prozeßdatenerfassung und -verarbeitung, industrielle Steuerungen und OEM-Anwendungen. Auch als Personal-Computer der absoluten Spitzenklasse sowie für Anwendungen in der kommerziellen mittleren Datentechnik ist SAM 68K bestens geeignet.

## 2.2 Technische Daten SAM 68K

Prozessor

* 68000 Prozessor mit interner 32-Bit-Struktur (achtzehn 32-Bit-Register)
* 16-Bit-Datenbus, 23-Bit-Adreßbus, 16 MB Hauptspeicher (voll dekodiert)
* 8 MHz Prozessortakt, abgeleitet aus einem 16 MHz Systemtakt
* sieben dekodierte Interrupt-Ebenen
* DMA- und multiprozessorfähig

Speicher

* 256 KB oder 640 KB dynamisches RAM auf der Grundplatine, extern erweiterbar bis ca. 16 MB
* zwei Sockel für EPROM's, per Jumper konfigurierbar für die Typen 2732, 2764, 27128, 27256. Das ergibt je nach Bestückung einen EPROM-Speicherbereich von 8, 16, 32 oder 64 KB

Floppy-Disk-Controller

* NEC 765 Controller-Baustein
* Shugart/BASF-Busbelegung zum Anschluss von bis zu vier Laufwerken (5" oder 8", single oder double Density)
* aufwendiger, voll integrierter Datenseparator (digitale PLL), benötigt keinen Abgleich
* Precompensation einstellbar

Video- und Grafikteil

* kompletter, Videoteil ("memory-mapped") für Schrift und hochauflösende Grafik (6845 Controller)
* Schrift:
  - 25 Zeilen mit je 80 Buchstaben
  - Groß/Kleinschreibung, echte Unterlängen
  - deutscher oder ASCII-Zeichensatz per Software wählbar
  - beliebige Zeichensätze und Schriftgrößen generierbar
* Grafik:
  - Auflösung per Software umschaltbar von 512 x 256 einzeln setzbaren Punkten auf 512 x 512 Punkte
  - 512 x 256 Punkte: non interlaced mode
  - 512 x 512 Punkte: interlaced mode
* Schrift und Grafik beliebig mischbar
* BAS-Ausgang zum Anschluß eines Video-Monitors
* paralleles Port für Tastaturanschluß

Ein/Ausgabe-Schnittstellen

* VIA 6522 frei für den Benutzer (zwei 8-Bit-Ports mit Handshake-Leitungen, zwei 16-Bit-Timer, Schieberegister)
* PIA 6821 für Tastaturanschluß (parallel) und Jumperfeld für Systemparameter (Mini/Standardfloppy, Auto-Boot, Terminalbetrieb u.a.)
* NEC 7201 Multi-Protokoll-Controller:
  - zwei V.24 Schnittstellen mit Treibern getrennt einstellbare Baudrate von 110 bis 19200 Baud
  - V.24 Schnittstellen wahlweise auch umschaltbar

auf schnelle synchrone Datenübertragung bis 800 KBaud mit SDLC, HDLC oder BiSync-Protokoll

* Alle Ein/Ausgabeleitungen auf Pfostensteckverbinder geführt

Businterface

* voll gepuffertes Businterface, voll kompatibel mit allen Baugruppen aus der EB 68000 Reihe von kws
* zwei 64-pol. Steckverbinder DIN 41612 Bauform C

Mechanische Daten

* doppelseitige, durchkontaktierte Leiterplatte im verlängerten Doppeleuropaformat (220 x 233.4 mm)
* alle ICs steckbar in hochwertigen, vergoldeten Präzisionsfassungen
* Einbau in genormte 19"-Baugruppenträger z.B. von Fa. SCHROFF möglich

Stromversorgung

* bei Betrieb ohne V.24 Schnittstellen nur +5 V Stromversorgung nötig
* Stromaufnahme

| | | | | |
|---|---|---|---|---|
| + 5 V | +- 5% | typ. | 2.5 A | (SAM-68K/256) |
| + 5 V | +- 5% | typ. | 3.2 A | (SAM-68K/640) |
| + 12 V | +-10% | max. | 0.2 A | |
| - 12 V | +-10% | max. | 0.2 A | |

## 2.3 Lieferumfang

Zum Lieferumfang von SAM 68K als Einzelplatine gehören:

a) Eine gedruckte Schaltung SAM 68K, bestückt in Abhängigkeit von den gewählten Optionen.

b) Ein 64pol. Steckverbinder (Federleiste) zum Anschluß der Versorgungsspannungen

c) Ein Prozessorhandbuch MC 68000

d) Das vorliegende Benutzerhandbuch

e) Ein Listing des Monitorprogramms

## 2.4 Systemkonfiguration

SAM 68K enthält auf einer Platine alle die Baugruppen, die man benötigt, um ein komplettes 68000-System aufzubauen.

Bild 2.1 zeigt, wie ein solches (Grund-)System normalerweise aussehen wird:

SAM 68K als zentrale Baugruppe, daran angeschlossen:

a) Tastatur mit parallelem Ausgang

b) Videomonitor mit BAS- oder TTL-Eingang als Datensichtgerät

c) Ein oder (besser) zwei Floppy-Disk-Laufwerke

d) Netzteil zur Versorgung dieser Baugruppen

e) Bei Bedarf Systemerweiterungen wie Drucker, Speichererweiterungen, Ein/Ausgabe-Erweiterungen usw.

Zu a): Es kann fast jede Standardtastatur mit parallelem TTL-Ausgang (8 oder 7 Bit mit Strobe) benutzt werden. Für Arbeiten mit dem Screen-Editor ist jedoch ein separates Cursorsteuerfeld mit Funktionstasten wünschenswert, aber nicht nötig. Das SUSY - Betriebssystem kann automatisch die Treiberroutinen laden, die für die verwendete Tastatur benötigt werden.

Zu b): Der Videomonitor sollte eine ausreichend hohe Bandbreite ( > ca. 18 Mhz) besitzen. Soll mit einer Grafikauflösung von 512 x 512 oder 768 x

480 Punkten gearbeitet werden, dann ist die Verwendung einer lang nachleuchtenden Bildröhre sehr empfehlenswert (P 39 Phosphor).

Zu c): Es können wahlweise 5" oder 8" Laufwerke angeschlossen werden. Beachten Sie jedoch, daß deren Speicherkapazität dem hohen Niveau dieses Rechners angepaßt sein sollte. Wir empfehlen deshalb:

bei 5" Disks: Doppelseitige 96-tpi-Laufwerke wie TEAC FD-55F mit ca. 650 KB Speicherkapazität pro Laufwerk

bei 8" Disks: z.B. BASF oder NEC Slimline-Laufwerke, doppelseitig mit ca. 550 KB Speicherkapazität pro Laufwerk

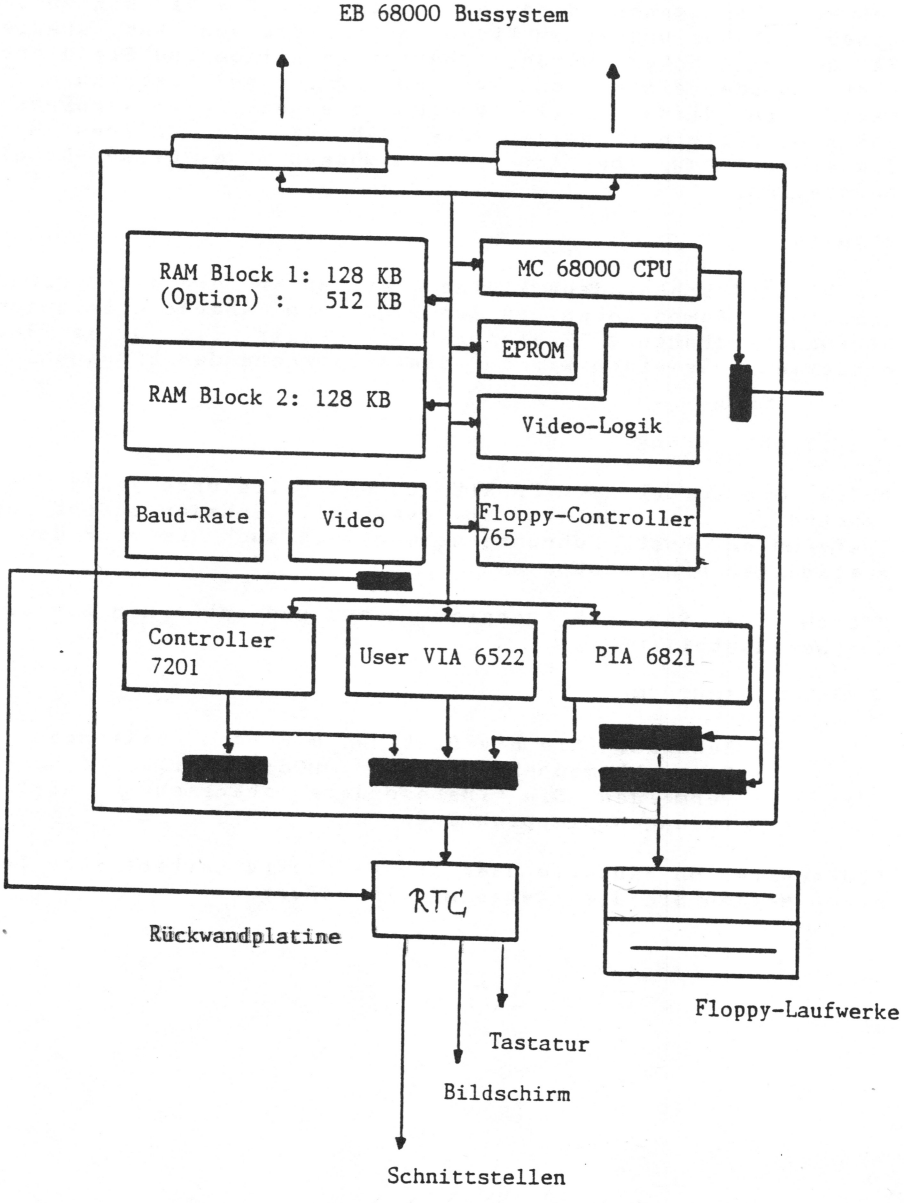
Zu e): Ein Drucker wird vorzugsweise über die V.24 Schnittstelle Nr. 2 angeschlossen. Es ist jedoch softwaremäßig auch eine Centronics Parallel-Schnittstelle implementiert, die einen der beiden VIA 6522 Ports benutzt.

Wie beim EB 68000 System von kws kann auch an SAM 68K ein Philips Mini-Digital-Cassettenrecorder direkt angeschlossen werden, sodaß ein System auch ohne Floppy-Disk-Laufwerke auskommen kann. Ein Teil der Software ist auf Kassetten erhältlich.

Per Jumper kann noch eine andere Systemkonfiguration ausgewählt werden, die ebenfalls von der Software voll unterstützt wird: anstelle einer Tastatur kann an die V.24 Schnittstelle Nr.1 ein normales Terminal angeschlossen werden, das dann z.B. als Bedienterminal dient, während für grafische Darstellungen der Videomonitor benutzt wird.

Als Option ist eine Rückwandplatine (RTC 1) erhältlich. Sie verteilt das Flachbandkabel, das von der SAM-68K-Platine kommt, auf einzelne Steckverbinder (MinD-Buchsen). Siehe auch Kapitel 9. Außerdem enthält die Platine eine batteriegepufferte Uhr, einen Leitungstreiber für die Centronics Schnittstelle sowie einen "Piepser".

Bild 2.1: Systemkonfiguration (Rückwandplatine optional)



# 3 Inbetriebnahme

## 3.1 Einführung

In diesem Kapitel sind alle Maßnahmen zusammengefaßt, die nötig sind, um SAM 68K als Einzelplatine zum ersten Mal in Betrieb zu nehmen. Wir gehen dabei davon aus, daß Sie SAM 68K zusammen mit einer Tastatur und einem Videomonitor, die von kws speziell für SAM 68K angeboten werden, betreiben wollen und Sie diese Geräte über kws oder einen von kws autorisierten Distributor bezogen haben. In diesem Falle wurden die benötigten Verbindungskabel schon mitgeliefert. Wollen Sie SAM 68K mit anderen Monitoren, Tastaturen usw. betreiben, so müssen Sie diese Kabel selbst herstellen.

WICHTIG:

kws bietet verschiedene kleine Platinen an, die die Verbindung der einzelnen Komponenten und den Einbau in Gehäuse erleichtern (z.B. Rückwandverdrahtung, Tastenplatine). Falls Sie diese Baugruppen einsetzen, verweisen wir auf die entsprechenden Einbauanleitungen.

## 3.2 Sichtprüfung

Prüfen Sie sofort nach Erhalt, ob das gelieferte Paket mechanisch beschädigt ist. Melden Sie eventuelle Schäden sofort sowohl dem Lieferanten (Post, Fuhrunternehmer) als auch kws oder dem für Sie zuständigen Distributor.

Packen Sie SAM 68K vorsichtig aus und entfernen Sie alle losen Verpackungsteile.

ACHTUNG:

Behandeln Sie die Platine mit den bekannten Vorsichtsmaßnahmen für MOS-Bauteile, d.h. vermeiden Sie insbesondere statische Aufladungen (Teppichböden!).

Prüfen Sie, ob Bauteile oder die gedruckte Leiterplatte beschädigt sind. Melden Sie dies gegebenfalls sofort.

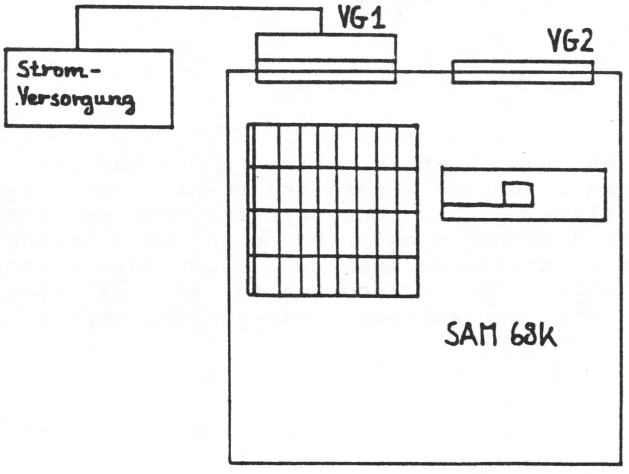
## 3.3 Vorbereitung des Systems

a) Vergewissern Sie sich, daß das von Ihnen verwendete Netzteil die geforderten Spannungen auch unter der angegebenen Belastung liefert und schließen Sie diese bei ausgeschaltetem Netzteil an die richtigen Pins des Steckverbinders VG1 (BEIM SPEICHER !) an (Bild 3.1):

+ 5V -> Pin # 31a, 31c, 32a, 32c
Masse -> Pin # 1a, 1c, 2a, 2c
+12V -> Pin # 3a
-12V -> Pin # 3c

Die Belegung der Busstecker ist, auch was die Versorgungsspannungen betrifft, unterschiedlich! Verwechseln Sie deshalb nie diese Busstecker. Eine falsche Polung der Versorgungsspannung kann den gesamten Rechner zerstören. Für diesen Fall lehnt kws jegliche Gewährleistungsansprüche ab.

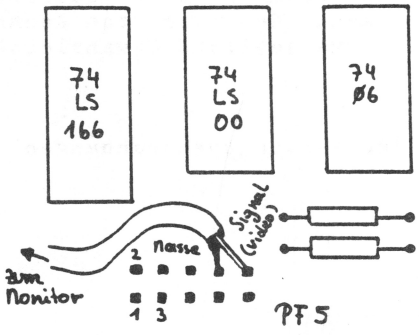
Bild 3.1: Anschluß der Versorgungsspannungen



b) SAM 68K besitzt viele Jumper (kleine Kurzschlußstecker), mit denen verschiedene Systemparameter eingestellt werden können. Diese Jumper sind bei der Auslieferung auf ihre Normalstellungen eingestellt. Prüfen Sie dies anhand des Jumperplanes in Kapitel 7 nach und führen Sie gegebenenfalls eine Korrektur durch. Die Jumper lassen sich mit einer kleinen Zange oder Pinzette leicht abziehen oder aufstecken.

c) Schließen Sie den Videomonitor gemäß Bild 3.2 an den Videoausgang von SAM 68K an.

Bild 3.2: Anschluß des Videomonitors




d) Testen Sie, ob bis hierher alles in Ordnung ist, indem Sie zunaechst den Helligkeits- und Kontrastregler des Videomonitors auf eine mittlere Einstellung bringen, dann den Videomonitor einschalten (Bild bleibt dunkel) und dann nach einer kleinen Aufwärmzeit des Monitors auch die Spannungsversorgung von SAM 68K einschalten. (Die Spannungen für SAM 68K müssen nicht in einer bestimmten Reihenfolge eingeschaltet werden.)

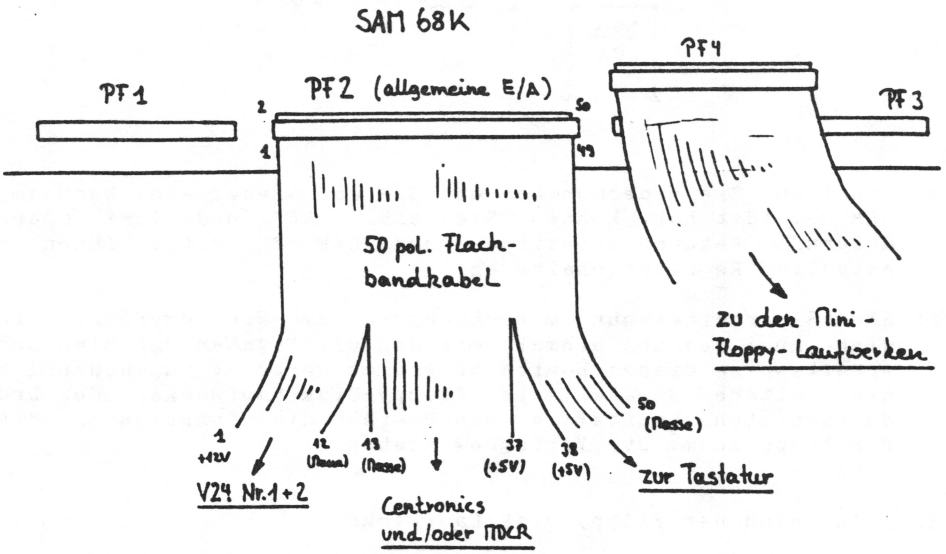
Jetzt erscheint auf dem Monitor die Meldung "SAM 68K MONITOR Vx.x" und ein Stern in der naechsten Zeile als Zeichen dafür, daß SAM 68K betriebsbereit ist. Stellen Sie gegebenenfalls das Bild auf einen angenehmen Helligkeits- und Kontrastwert ein.

Sollte sich SAM 68K nicht in dieser Weise melden, so schalten Sie zunaechst wieder alles ab und vollziehen die Schritte bis hierher nochmals nach.

e) SAM 68K wartet jetzt auf eine Eingabe von Ihnen, die Sie jedoch noch nicht geben können, da noch keine Tastatur angeschlossen ist. Schalten Sie deshalb erst einmal wieder alles aus.

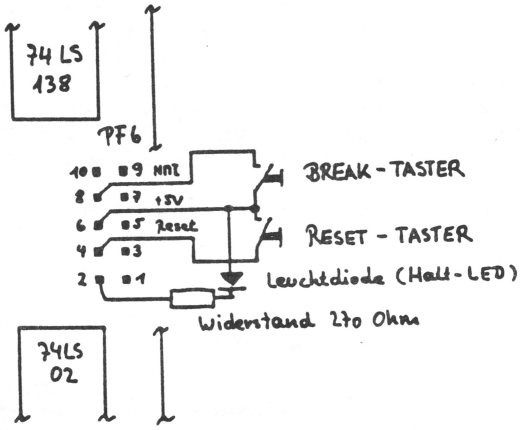
f) Schließen Sie die Tastatur an den 50-poligen Ein- und Ausgabesteckverbinder PF2 gemäß Bild 3.3 und Tabelle 7.3 an. Wenn Sie eine Kabelverteilerplatine mitbestellt haben, so finden Sie hierzu eine separate Anleitung. Achten Sie darauf, daß Tastaturen, die nicht von kws bezogen wurden, in aller Regel eine andere Anschlussbelegung haben, die Sie zunächst auf die Steckerbelegung von SAM 68K umrüsten muessen!

Bild 3.3: Anschluß der Peripheriegeräte



g) Schließen Sie gemäß Bild 3.4 einen RESET- und BREAK-Taster an den Pfostensteckverbinder PF6 an. Diese Taster sind nicht unbedingt nötig, aber oft praktisch. RESET- und BREAK-Taster können Sie dazu benutzen, um SAM 68K in einen definierten Ausgangszustand zu versetzen. Betätigen Sie aber niemals diese Schalter, wenn der Rechner z.B. auf Floppy-Disk-Laufwerke zugreifen will. Dies kann den Inhalt einer Diskette unwiderruflich zerstören! An diesen Pfostensteckverbinder können Sie auch eine Leuchtdiode (mit Widerstand!) anschließen, die Ihnen anzeigt, wenn der Prozessor im HALT-Zustand ist.

Bild 3.4: Anschluß des Reset- und Break-Tasters




h) Schalten Sie Videomonitor und SAM 68K wieder ein. Nachdem sich SAM gemeldet hat, können Sie z.B. "RE" und <cr> (Carriage Return, Return) eintippen und SAM 68K zeigt Ihnen seine aktuellen Registerinhalte an.

i) SAM 68K arbeitet nun im Monitorprogramm. Wir empfehlen Ihnen, sich nun zuerst einmal mit den wichtigsten Befehlen und der Arbeitsweise dieses Monitorprogramms vertraut zu machen, bevor sie weitere Geräte, wie Floppy-Disk-Laufwerke oder Drucker dazuschalten. Kapitel 4 beschreibt die Funktionen, die im Monitorprogramm zur Verfügung stehen.

## 3.4 Anschluß der Floppy-Disk-Laufwerke

Laufwerke, die direkt über kws für SAM 68K bezogen wurden, sind für dem Anschluß schon entsprechend vorbereitet.

Gehen Sie in diesem Fall folgendermaßen vor:

Schließen Sie die Versorgungsspannung der Drives an. Achten Sie

dabei auf ausreichend dicke Kabel. Schließen Sie sie über ein mit den entsprechenden Steckern versehenes Flachbandkabel an den Steckverbinder PF4 (bei 5" Floppys) oder PF3 (bei 8" Floppys) an. Prüfen Sie anhand des Jumperplanes, ob die Jumper, die den Floppy-Teil von SAM 68K betreffen, richtig stehen (Jumper J6 4-11, BR17, BR18, BR19).

Achten Sie darauf, daß nur auf einem Laufwerk die Abschlußwiderstände für den Floppy-Bus installiert sein dürfen und die Drives über die Leitungen DRIVE SELECT 0 und 1 ansprechbar sind.

Wenn Sie Laufwerke, die nicht über kws bezogen wurden, anschließen wollen, so beachten Sie, daß die auf SAM 68K installierte Hard- und Software folgende Anforderungen an die Drives stellt:

- Auswahl der Drives über die einzelnen Select-Leitungen. (RADIAL SELECT, nicht "BINARY")

- Stepper Motor sollte stets aktiviert sein, nicht nur im Zusammenhang mit z.B. HEADLOAD

- Zweiseitige Laufwerke können nicht als zwei unterschiedliche Laufwerke angesprochen werden.

- HEADLOAD sollte über die HEADLOAD-Steuerleitung allein oder über HEADLOAD und DRIVE SELECT gesteuert werden. 5" Laufwerke, bei denen die Möglichkeit besteht, den Kopf auch in einem Pre-Ready Zustand zu laden, sollten in dieser Weise betrieben werden (z.B. HM-Jumper bei TEAK FD 55F).

- Auswahl der Diskettenseite durch SIDE-SELECT Eingang

- Betriebsart ist SOFT-SECTORED

- Step-Richtung muß über DIRECTION und STEP gesteuert werden

- Besondere Filter und Schaltkreise zur Datenaufbereitung auf den Floppy-Laufwerken sollten desaktiviert sein

## 3.5 Anschluß eines Druckers

Ein Drucker sollte vorzugsweise über die V.24 Schnittstelle Nr.2 an SAM 68K angeschlossen werden. Es ist jedoch auch eine einfache Centronics Schnittstelle softwaremäßig implementiert, die ein Port der VIA 6522 benutzt. In diesem Falle erfolgt die Übertragung mit 7 Bit parallel. Diese Leitungen sind jedoch nicht gepuffert, sodaß die Leitungslänge ca. 2 m nicht überschreiten sollte. Auf der als Zubehoer erhaeltlichen Kabelplatine befinden sich Puffer fuer die Centronics Schnittstelle; es können dann auch 8 Bit uebertragen werden.

## 3.6 Datenformat der V.24-Schnittstellen

Das Monitorprogramm initialisiert die V.24-Schnittstellen fuer folgendes Uebertragungsformat:

1 Startbit, 8 Datenbits, kein Paritybit, 1 Stopbit

Dieses Datenformat wird auch von den meisten Betriebssystemen verwendet, die auf dem SAM-68K lauffaehig sind. Es können jedoch per Software auch andere Datenformate gewaehlt werden. Wir verweisen diesbezüglich auf das Datenblatt des MPSC-Controllers und die Initialisierungsroutinen im Monitorprogramm.

# 4 Monitorprogramm von SAM 68K

## 4.1 Einführung

Das Monitorprogramm ("der Monitor") befindet sich in den beiden EPROMs. Neben den notwendigen Initialisierungen nach einem (Power-on)-Reset bietet er alle Funktionen, die nötig sind, um ein in Maschinensprache geschriebenes Programm einzulesen, zu starten, auszutesten und abzuspeichern. Darüber hinaus verwaltet er verschiedene Ein/Ausgabe-Kanäle und stellt diese Treiberroutinen auch dem Benutzer zur Verfügung.

Weiterhin sorgt der Monitor dafür, daß verschiedene Betriebssysteme von der Floppy geladen werden können und unterstützt diese bei der Ein- und Ausgabe.

Ein wichtiger Teil des Monitors sind die Routinen, die die Tastatur und die Ausgabe auf dem Bildschirm steuern und verwalten. Man nennt solche Routinen "Treiberroutinen" oder "Treiber".

Der Monitor besitzt für Anwenderprogramme einen Eintrittspunkt für einen "Warmstart". Beim Reset wird der Monitor automatisch bei seinem Kaltstart gestartet. Kalt- und Warmstart unterscheiden sich durch die vorgenommenen Initialisierungen. Beim Kaltstart werden prinzipiell alle Systemvariablen auf ihre Anfangswerte gesetzt, Breakpoints gelöscht und die Ein/Ausgabe wird über die Tastatur und den Videoteil abgewickelt, sofern nicht per Jumper auf Terminal-Betrieb umgestellt wurde.

Nach den Start befindet sich der Monitor in der Haupt-Kontroll - Schleife (HKS) und versucht, eingegebene Kommandos auszuführen.

## 4.2 Versions- und Revisionsnummer

Nach dem Einschalten der Versorgungsspannungen oder dem Drücken der Reset-Taste meldet sich der Monitor mit

```
SAM 68K MONITOR Vx.y
*
```

Dabei ist x die Versions- und y die Revisionsnummer. Bitte geben Sie diese Nummern bei allen Rückfragen an.

Unterschiedliche Versionsnummern bedeuten, daß sich die Monitore in ihrer Funktionsweise deutlich unterscheiden, d.h., daß z.B. neue Kommandos oder Treiberroutinen aufgenommen wurden oder der Monitor für spezielle Anwendungen adaptiert wurde. Bei unterschiedlichen Revisionsnummern ist die Funktionsweise des Monitors gleich, jedoch können z.B. eventuelle Fehler verbessert worden sein.

In beiden Fällen können sich die absoluten Adressen im

Monitorprogramm ändern. Der Benutzer sollte deshalb eigene Programme möglichst unabhängig vom Monitor schreiben. Für häufig vorkommende Routinen ist jedoch eine Sprungtabelle vorgesehen, deren Lage bei allen unterschiedlichen Revisionen und den meisten unterschiedlichen Versionen erhalten bleibt.

## 4.3 Format der Kommandozeile

Kommandos für den Monitor werden so eingegeben, wie dies für pufferorientierte Computersysteme üblich ist. Die Interpretation einer Kommandozeile beginnt erst, wenn <cr> (Carriage Return) gedrückt wird. Innerhalb einer Zeile kann mit der "Delete"-Taste oder durch Eingabe von Control 'H' (Backspace) das letzte Zeichen gelöscht werden.

Erzeugt ein Kommando längeren Output auf der Konsole, so kann dieser durch Drücken der Leertaste in vielen Fällen angehalten werden. Anschließende Eingabe von <cr> bricht das Kommando ab, Eingabe eines beliebigen anderen Zeichens setzt den Output fort.

Das Format einer Kommando-Zeile ist immer:

*COmmando <parameter>

dabei bedeuten:

| | |
|---|---|
| * | Prompt des Monitors. Wird vom Computer ausgegeben. |
| CO | Eingabe, die notwendig ist, damit der Monitor das Kommando erkennt |
| mmando | Eingabe, die nicht notwendig, aber zulässig ist. |
| <parameter> | Spezielle Parameter, abhängig vom jeweiligen Kommando |

In eckigen Klammern eingeschlossene Parameter sind nicht unbedingt erforderlich. Wenn sie angegeben werden, überschreiben sie einen Default-Wert.

Im folgenden bedeutet "reg#" eines der 68000-Register in der folgenden Syntax:

| | |
|---|---|
| D0, D1, D2, D3, D4, D5, D6, D7 | Datenregister |
| A0, A1, A2, A3, A4, A5, A6, A7 | Adressregister |
| PC | Programmzähler |
| SR | Statusregister |
| SS | Supervisor Stackpointer |
| US | User Stackpointer |

Bei der Eingabe von Zahlen können führende Nullen weggelassen werden.

Beispiel:

| | |
|---|---|
| *REGISTERS | Zeige Register an |
| *RE | Dasselbe in Kurzform |
| *GO 1B00 | Starte Programm bei Adresse 1B00 |
| *GO | Starte Programm beim aktuellen PC-Wert |

## 4.4 Register anzeigen und ändern

| Kommando | Beschreibung |
|---|---|
| reg# | Zeige Inhalt des angegebenen Registers an |
| reg# = <hexdata> | Lade den hexadezimalen Wert in das angegebene Register |
| REgisters | Zeige den Inhalt aller Register an, die in der Registerliste aktiviert sind |
| RList | Definiere aktive Register in der Registerliste |

Auf das Kommando "RL" hin werden zunächst die Namen aller Register ausgegeben (2 Zeilen). Darunter befindet sich eine Zeile, in der durch ein "+" markiert ist, welches Register aktiviert ist. Die Inhalte dieser aktivierten Register werden durch das RE Kommando angezeigt.
Nun wartet der Monitor auf eine Eingabezeile. Soll die Registerliste geändert werden, so gibt man hier für jedes Register, welches aktiviert werden soll, ein "+", für jedes, das desaktiviert werden soll, ein "-" ein. Eingabe von <cr> gleich am Anfang der Zeile läßt die Registerliste unverändert. Man lese die Registerliste senkrecht von oben nach unten.

| Beispiel | Beschreibung |
|---|---|
| *D0=123 | Lade Datenregister 0 mit dem hexadezimalen Wert 123 |
| *D0 | Zeige den Inhalt von D0 an |
| D0 = 00000123 | |
| *A0=FFF | Lade A0 mit FFF |
| *PC=1B00 | Lade den Programmzähler mit 1B00 |
| *RL | Zeige Registerliste an |
| DDDDDDDDAAAAAAASUSP | |
| 01234567012345 6SSRC | |
| +++++++++++++++++++ | Anfänglich sind alle Register aktiviert |
| +-------+---------+ | Nur D0, A0 und PC anzeigen |
| *RE | Zeige Inhalt der aktivierten Register an |
| D0 = 00000123 A0 = 00000FFF PC = 00001B00 | |
| * | |

## 4.5 Speicherzellen anzeigen und ändern

| Kommando | Beschreibung |
|---|---|
| DUmp <start> <end> | zeigt in tabellarischer Form den Inhalt des Speicherbereiches an, der zwischen der angegeben Start- und Endadresse liegt. Am Anfang der Zeile steht die erste Adresse, dann folgen 16 Bytes Speicherinhalt und schließlich die ASCII Darstellung dieser Zeichen. Nichtdruckende ASCII-Zeichen werden durch einen Underscore dargestellt. |
| FI <A1> <A2> <VAL> | füllt den Speicherbereich von Adresse A1 bis A2 mit dem Hex. Wert VAL |
| MO <A1> <A2> <TO> | schreibt den Inhalt des Speicherbereiches von A1 bis A2 in den Bereich ab TO. Die Bereiche dürfen sich nicht überlappen. |
| OPen <hexaddress> | öffnet die angegebene Adresse zum Lesen oder Schreiben. Defaultadresse ist 0. Die Adressen dürfen nur gerade sein, ungerade werden abgerundet. Angezeigt und geändert wird stets der 16-bit Inhalt der Adresse. |

In der "OP"-Routine gibt es folgende Unterkommandos:

| | |
|---|---|
| ↑ | öffnet die ein Wort niedrigere Adresse |
| <cr> | öffnet die ein Wort höhere Adresse |
| =<hexdata> | lädt den hexadezimalen Wert in die Speicherzelle |
| #<hexaddress> | öffnet die angegebene Adresse |
| Q | beendet die "OP"-Routine |

Syntaxfehler werden durch ein "?" angezeigt und die Adresse wird nochmals angegeben. Versucht man, eine Adresse zu beschreiben, die danach nicht den angegebenen Wert enthält (EPROM, ACIA, ...), so wird der Wert angezeigt, den diese nach dem Änderungsversuch hat und danach ein "?" ausgegeben.

| Beispiel | Beschreibung |
|---|---|
| *OP 1B00 | öffne Adresse 1B00 |
| 001B00 1234 =4142 | lade 1B00 mit hex. 4142 |
| 001B02 1234 ↑ | zurück |
| 001B00 4142 <cr> | vorwärts |
| 001B02 1234 Q | verlasse "OP"-Routine |

```
*DUMP 1B00 1B0E      generiere Speichertabelle
001B00 41 42 12 34 00 00 00 00 00 00 00 00 00 00 00 00 AB..
*FI 2000 2100 00     schreibt $00 in den Bereich von $2000 bis
*                    2100
```

## 4.6 Starten und Testen eines Programms

| Kommando | Beschreibung |
|---|---|
| GO <hexaddr.> | Startet ein Programm (JMP) bei der angegeben Adresse; wird diese nicht angegeben, dann bei der Adresse, auf die der PC zeigt (z.B. nach TR). Aktiviert alle Breakpoints. |
| CAll <addr> | Wie "GO", jedoch Aufruf als Subroutine (JSR). |
| STep | Ein Single Step wird ausgeführt und danach der Inhalt der Register angezeigt, die in der Registerliste aktiviert sind. Ein Single Step ist auch immer dann nötig, wenn ein Programm zuvor auf einen Breakpoint aufgelaufen ist. Daraufhin kann das Programm mit ST, TR oder GO weiter fortgesetzt werden. |
| TRace | Der Trace-Mode wird eingeschaltet und nach jedem ausgeführten Befehl der PC und alle aktivierten Register angezeigt. Der Trace-Mode kann durch Betätigen der Leertaste angehalten und wieder fortgesetzt werden. Ein <cr> beendet den angehaltenen Trace-Mode, es kann dann mit ST oder GO fortgefahren werden. |
| TD | (Trace Display formatted). Vor jedem neuen Registerdump wird ein Form Feed ($0C) gesendet. |
| TC | (Trace Display Clear). TD wird abgeschaltet. |
| Bn=<hexaddr.> | Einer der acht Breakpoints (n=0-7) wird auf die angegebene Adresse gesetzt und beim Starten des Programms aktiviert. Einzelne Breakpoints können durch Nullsetzen (Bn=0) gelöscht werden. |
| CB | Löscht alle Breakpoints (Clear Breakpoints) |
| DB | Eine Liste der Breakpoints wird ausgegeben (Display Breakpoints) |
| Bn | Zeigt die Adresse des Breakpoints n an |

| Beispiel | Beschreibung |
|---|---|
| *B0=1B04 | Setze Breakpoint auf Adresse 1B04 |

```
*B0                        Zeige Breakpoint 0 an
B0 = 00001B04
*GO 1B00                   Starte Programm bei Adresse 1B00
BREAK AT $00001B04         Breakpoint erreicht
*ST                        Single Step
D0 = 00000022  A0 = 00000113  PC = 00001B06
*CB                        Lösche Breakpoints
*GO                        Weiter im Programm
```

## 4.7 Ein/Ausgabe-Kanäle

| Kommando | Beschreibung |
|---|---|
| IO | Definiere Ein/Ausgabekanäle |

Das Monitorprogramm gestattet es, jeweils 16 verschiedene Ein/Ausgabekanäle zu verwalten.

Auf das Kommando IO hin wird zunächst eine Liste dieser Kanäle ausgegeben. I0 bis IF sind die Input-Kanäle 0 bis 15, O0 bis OF die Output-Kanäle 0 bis 15. Aktive Kanäle werden durch ein "+", nicht aktive Kanäle durch ein "-" gekennzeichnet. Das Monitorprogramm wartet nun auf eine Eingabezeile. Soll die Liste der aktiven Kanäle geändert werden, so gebe man unter jedem zu aktivierenden Kanal ein "+", unter jedem zu desaktivierenden Kanal ein "-" ein. <cr> gleich am Anfang der Zeile läßt die Liste unverändert.

Im Augenblick werden vom Monitor unterstützt:

```
Kanal  0: Ein/Ausgabe über Konsolkanal
Kanal  1: Ein/Ausgabe über V.24 Nr. 2
Kanal  2: Ein/Ausgabe über MiniDCR Bandlaufwerk
Kanal  3:     Ausgabe über Centronics Schnittstelle
Kanal  4:     Ausgabe über Bildschirm
Kanal  5:     Ausgabe über V.24 Nr. 1
Kanal 15: Ein/Ausgabe über vom Benutzer definierten
          Kanal
```

Der Konsolkanal wird standardmäßig für alle Ein/Ausgaben benutzt. Je nach Stellung des Jumpers J6 5-12 wird dazu entweder Tastatur/Bildschirm oder die V.24-Schnittstelle Nr.1 (für den Betrieb mit Terminal) benutzt.

Kanal 15 ist für die Benutzer vorgesehen, die eigene Treiberroutinen implementieren wollen. Diese werden über einen Vektor im RAM angesprungen. Näheres dazu finden Sie im Listing des Monitorprogramms.

Nach einem Reset erwartet der Monitor stets Ein/Ausgabe von der Konsole. Konsole ist Tastatur/Bildschirm oder, wenn der Jumper entsprechend gesetzt ist, die V24-Schnittstelle Nr.1. In beiden Fällen ist die Eingabe dabei interrupt-gesteuert.

Während mehrere gleichzeitig aktivierte Inputkanäle in der Regel nicht benötigt werden, kann eine Konfiguration mit zwei oder mehreren aktiven Outputkanälen durchaus vorkommen. Z.B. ist es auf diese Weise leicht möglich, ein Druckerprotokoll von der Arbeit am Terminal anzufertigen.

Beispiel:

```
*IO
OOOOOOOOOOOOOOOOIIIIIIIIIIIIIIII
0123456789ABCDEF0123456789ABCDEF
+---------------+---------------  Anfänglich ist Konsole aktiv
++--------------+---------------  Zusätzliche Ausgabe auf
                                  V.24 Nr.2 (z.B. Drucker)
```

Beschreibung der Centronics Schnittstelle

Der Centronics Output Treiber ist bei gesetztem I/O-Flag für Kanal #3 über die "OUTCH" Routine für Benutzerprogramme ansprechbar oder er wird vom Monitorprogramm automatisch mitbenutzt.

Die Übertragung erfolgt über 7 parallele Datenleitungen mit DATASTROBE (negativer Puls mit min. 3 us Breite) und unter Beachtung der BUSY Leitung (HI = Busy).

Belegung der parallelen E/A-Leitungen falls MiniDCR und/oder Centronics Schnittstelle benutzt werden

siehe Tabelle in 7.3

## 4.8 Laden und Abspeichern von Programmen

a) Laden von Programmen

Kommando:

LOad <start address>

Der Monitor fragt nun "FILE DEFINITION?". Diese muss lauten:

"[<name>]" [#<incha>] [/"<message>"#<outcha>]

Beschreibung:

Mit dem LOAD-Kommando können Motorola-kompatible S1/S2-Records in den Speicher geladen werden. Im S0-Record steht gegebenenfalls der Name des Files, Ende-Kennung ist S8 oder S9. Die Checksumme wird

immer überprüft, ein Fehler führt zum Abbruch. Dieser Befehl erlaubt es, von beliebigen Kanälen zu laden und gegebenfalls auf einem Kanal eine Nachricht zu senden (z.B. an ein Hostsystem).

Zur Syntax:

Fehlt <start address> so wird auf die im File angegebenen Adressen geladen. Ist <start address> angegeben, so wird das File relokativ, beginnend mit der Startadresse geladen.
Wird <name> angegeben, so wartet der Treiber auf ein File mit diesem Namen und lädt nur dieses. Wird zwischen den Anführungszeichen kein Name angegeben, so werden automatisch alle ankommenden S1/S2-Records bis zum nächsten S8/S9-Record geladen.
#<incha> ist der Kanal, von dem geladen werden soll (0...F); fehlt <incha>, so wird der Kanal #2 (MDCR) benutzt.

Beispiel:

```
*LOAD                    Liest das File "PROGRAM1" vom MDCR
FILE DEFINITION?
"PROGRAM1"
```

Weiterhin gestattet es der LOAD-Befehl, vor dem Laden auf einem beliebigen Out-Kanal eine Nachricht zu senden. "<message>" ist diese Nachricht, #<outcha> (0...F) ist der Kanal, auf dem sie gesendet werden soll. Damit kann z.B. ein Hostsystem veranlasst werden, ein File zu senden, das dann von SAM 68K geladen wird.

Beispiel:

```
*LO 1000
FILE DEFINITION?
"PROGRAM1"#1/"LIST PROGRAM1.TXT"#1
```

Veranlaßt das an V.24 Nr.2 angeschlossene Hostsystem, das Textfile PROGRAM1 auszugeben. Dieses wird von der EB 68000 CPU ab der Adresse $1000 geladen. Das Textfile muß natürlich Code im Motorola-Format enthalten.

b) Abspeichern von Programmen

Kommando:

SAve <startadress> <endadress>

Beschreibung:

Der Speicherbereich zwischen der Start- und Endadresse wird in Motorola-kompatiblen S1/S2-Records ausgegeben. Auf dieses Kommando hin erwartet der Monitor in einer zweiten Eingabezeile die zugehoerige Filedefinition. Diese muss lauten:

"[<name>]"[#<outcha1>][/"<message>"#<outcha2>]

Dabei ist <name> der Name, den das File erhalten soll und <outcha1> (0...F) der Outkanal, auf den gesendet werden soll. <name> ist optionell; fehlt #<outcha>, so wird der Kanal #2 (MDCR) benutzt.

Wiederum ist es möglich, zuvor auf dem Outkanal <outcha2> eine Nachricht zu senden (siehe "LOAD"-Kommando).

Beispiel:

| | |
|---|---|
| *SAVE 1000 13FF<br>FILE DEFINITION?<br>"PROGRAM1" #1 | Sendet den Speicherbereich von 1000 bis 13FF an V.24 Nr.2 und gibt dem File den Namen PROGRAM1 |
| *SAVE 1B00 1C00<br>FILE DEFINITION?<br>"TEST.BIN" | Speichert den angegebenen Bereich unter dem Namen "TEST.BIN" auf dem MDCR ab. |

## 4.9 Booten des Betriebssystems

| Kommando | Beschreibung |
|---|---|
| BOot | Startet das Floppy-Disk Betriebssystem, sofern SAM 68K in einem solchen System eingesetzt wird. |

## 4.10 Abbrechen eines Programms

Die Tastatur und die V.24 Nr.1 sind Interrupt-getrieben. Das Statusregister des Prozessors ist auf $2200 gesetzt, sodaß Interrupts ab Level 3 erlaubt sind. Die zu Level 3 gehörende Interruptroutine leitet alle Interrupts, die nicht von der Tastatur oder der V.24 Nr. 1 kommen, in das RAM. Standardmäßig finden sie dort ein RTE, können jedoch beliebig weitergeleitet werden (siehe auch Listing des Monitorprogramms).

Die Serviceroutine für den Input-Interrupt gestattet es, vom Terminal aus Programme abzubrechen, sofern nicht der Interrupt maskiert wurde oder gerade ein Interrupt mit höherer Priorität bedient wird. Während zeitkritischer Aufgaben, wie z.B. MDCR- oder Floppy-Steuerung, wird der Interrupt des Konsolkanals gesperrt, kann also nicht stören.

Abgebrochen wird ein Programm durch Senden von Control 'C'. Daraufhin meldet sich die Master Control Routine (MCR). Wird jetzt "ABO" eingegeben, so erfolgt Abbruch des Programms mit einem

Registerdump. Alle anderen Eingaben werden ignoriert und das Programm läuft weiter. Nach dem Programmabbruch können dann z.B. mit "RE" die Registerinhalte geprüft werden.

Beispiel:

```
(control 'C')
MCR>ABO<cr>
BREAK AT $1006    (es folgt ein Registerdump...)
*
```

## 4.11 Monitorroutinen für den Benutzer

Die in Kapitel 4.7 beschriebenen Ein/Ausgabe-Kanäle können auch von Benutzerprogrammen angesprochen werden. Hierfür ist ein Flag im RAM-Bereich und eine Sprungtabelle zu den wichtigsten Monitorroutinen vorgesehen.

Die 2 x 16 Ein/Ausgabekanäle werden unter der Adresse "IOFLAG" verwaltet. Das dort stehende 32-Bit-Wort ist mit seiner unteren Hälfte den Outputtreibern, mit seiner oberen Hälfte den Inputtreibern zugeordnet. Ein auf "1" gesetztes Bit aktiviert den entsprechenden Kanal, ein auf "0" gesetztes desaktiviert ihn. Als Standard wird beim Kaltstart des Monitors $0001 0001 gesetzt. Dies bedeutet, daß alle Ein/Ausgabe über die Konsole abgewickelt wird. Die Zuordnung der einzelnen Bits zu den Treibern ist also folgendermaßen (vergl. Kapitel 4.7):

```
                Eingabe                  Ausgabe
      Bit 31                    16 15                    0
         !-----------------------!----------------------!
 IOFLAG  ! !                   ! ! !                  ! !
         !-----------------------!----------------------!
```

| Eingabe über | Bit# | Ausgabe über | Bit# |
|---|---|---|---|
| USER def. | 31 | USER def. | 15 |
| | | V.24 Nr. 1 | 5 |
| | | Bildschirm | 4 |
| | | Centronics | 3 |
| MiniDCR | 18 | MiniDCR | 2 |
| V.24 Nr. 2 | 17 | V.24 Nr. 2 | 1 |
| Konsole | 16 | Konsole | 0 |

Sollen vom Benutzer definierte Treiber verwendet werden, so müssen die entsprechenden Anfangsadressen dieser Treiber in USEROUT+2 bzw. USERIN+2 eingetragen werden. Der Benutzer sollte kein Register verändert zurückgeben, außer in D0 (in D0.B wird die Information übergeben). Die Routinen müssen mit einem RTS enden.

Nachdem "IOFLAG" entsprechend gesetzt wurde, sollte über die Sprungtabelle des Monitors die gewünschte Subroutine angesprungen

werden. Die Sprungtabelle mit Erklärungen finden Sie im Listing des Monitorprogramms.

Es ist darauf zu achten, daß vor dem Wiedereintritt in den Monitor "IOFLAG" wieder auf die gewünschte Konfiguration gebracht wird!

Außerdem besteht noch die Möglichkeit, einen Bufferbereich mittels eines TRAP 15 Aufrufes ein- oder auszugeben.

Aufruf:

```
TRAP 15
DC.W <function>
```

Beschreibung:

<function> = 1 : Einlesen in einen Buffer. Vor dem Aufruf dieser Funktion müssen (A5) und (A6) auf den Anfang des Buffers zeigen. Danach zeigt (A6) auf das letzte eingelesene Zeichen.

<function> = 2 : Ausgabe eines Bufferinhaltes (A5) zeigt auf den Anfang, (A6) auf das Ende des Buffers.

Beidesmal werden die in "IOFLAG" aktivierten Treiber verwendet.

## 4.12 Video- und Tastaturtreiber

Interessierte Benutzer können die Funktionsweise der Video- und Tastaturtreiber aus dem Listing des Monitorprogramms entnehmen. Die Spezial-Funktionen, die die Videotreiber erfüllen können - zum Beispiel Umschaltung auf deutschen Zeichensatz u.ä. - werden durch Control-Zeichen oder Escape-Sequenzen initiiert. Eine Zusammenstellung dieser Möglichkeiten finden Sie im Kapitel 7 dieses Handbuches.

Die Tastaturtreiber codieren die Zeichen, die von der Tastatur kommen so um, daß sie der "kws-Norm" entsprechen. Wenn Sie eine spezielle Tastatur anschließen oder die Zuordnung der Tasten bei der Standard-Tastatur ändern möchten, so können Sie diesen Treiber entsprechend ändern. Am einfachsten geschieht dies, indem Sie unter dem SUSY-Betriebssystem beim Booten des Systems diese neuen Treiber dazubinden (STARTUP-File!).

Wichtig ist jedoch, daß Sie die Zeichen, die für eine spezielle Funktion (z.B. CURSOR hoch) vom Tastaturtreiber w e i t e r - gereicht werden, NICHT verändern. Es erwarten nämlich alle von kws gelieferten Programme diese Codes!

# 5 Kopplung mit Hostsystemen

## 5.1 Einführung

Ein vorhandenes Hostsystem kann SAM 68K als externer Massenspeicher dienen. Läuft auf diesem Hostsystem ein 68000 Crossassembler/Compiler, so kann die Programmentwicklung auf dem Hostrechner erfolgen und der Objektcode nach Fertigstellung von SAM 68K geladen und ausgeführt werden.

## 5.2 Praktische Durchführung

Das Hostsystem muß zu diesem Zweck eine V.24 Schnittstelle besitzen und sollte die Handshake-Leitungen RTS und CTS bedienen. Es wird an die V.24 Schnittstelle Nr. 2 von SAM 68K angeschlossen.

Für diese Kopplung stellt das Monitorprogramm einen speziellen Befehl zur Verfügung:

Kommando: HOst

Beschreibung:

Nach diesem Aufruf wird Input von der Tastatur auf der V.24 Nr.2 ausgegeben. Input von der V.24 Nr.2 wird auf dem Bildschirm ausgegeben. Man ist mit dem Hostsystem an V.24 Nr.2 "transparent" verbunden. Die Handshakeleitungen sollten bedient werden.

Nun kann man direkt auf das gesamte Betriebssystem des Hostrechners zugreifen und z.B. ein dort editiertes Programm mittels eines 68000 Crossassemblers/Compilers übersetzen.

Der Transparent-Mode wird abgeschaltet durch Senden von Control 'A' ($01). Danach befindet man sich wieder in der HKS des Monitors.

Jetzt kann z.B. das (in Motorolaformat vorliegende) 68000-Programm geladen werden. Eine typische Befehlssequenz wäre dann:

```
LOAD ""#1/"LIST CROSSFILE"#1
```

Für diese einfachen Anwendungen sollte keine spezielle Koppelsoftware auf dem Hostrechner erforderlich sein. Bedient er jedoch die Handshake-Leitungen nicht, so kommt dann nur ein einfacher Betrieb, z.B. das Senden von Crossfiles in Betracht.

# 6 Hardware-Beschreibung

## 6.1 Einführung

Dieses Kapitel enthält alle nötigen Informationen zur Hardware von SAM 68K.

## 6.2 Reset- und Break-Schaltung

Sie dient dazu, das System in einen definierten Ausgangszustand zu bringen. Dies geschieht automatisch beim Einschalten der Stromversorgung oder kann manuell durch Drücken der RESET Taste erreicht werden. Zu diesem Zweck ist ein Timer (556) vorgesehen, der als Monoflop geschaltet ist und der für die Dauer von etwa 2 sec die RESET und HALT Leitung Low zieht. Dadurch werden gleichzeitig die CPU, sowie die Peripheriebausteine initialisiert. Der Inhalt der RAM's wird dadurch nicht geändert.

Video- und RAM-Controller werden durch die RESET-Leitung nicht beeinflußt. Sie besitzen eine eigene RESET-Schaltung, die nur beim Einschalten der Versorgungsspannung aktiv wird.

Durch Drücken der BREAK Taste wird ein zweites, anderes Monoflop getriggert, das einen Autovector-Interrupt auf Ebene 7 auslöst und für ca. 100 msec jeden Interrupt sperrt. Diese Taste kann zum Abbrechen eines laufenden Programms benutzt werden.

## 6.3 Speicher

Auf der Platine sind je nach Konfiguration entweder 256 KB (bei SAM-68K/256) oder 640 KB (bei SAM-68K/640) dynamisches RAM sowie Fassungen für max. 64 KB EPROM untergebracht.

Der RAM-Bereich von SAM 68K ist immer in zwei Blöcke aufgeteilt. Block 1 beginnt immer bei der Adresse $00000 und endet beim SAM-68K/256 bei der Adresse $1FFFF und beim SAM-68K/640 bei der Adresse $7FFFF. Die Groesse des Blocks 1 ist im ersten Fall also 128 KB, im zweiten Fall dagegen 512 KB. Block 1 steht voll dem Prozessor zur Verfügung. Durch den intelligenten Refresh - Controller wird bei jedem Zyklus, der nicht diese RAM-Bank anspricht, ein versteckter Refresh durchgeführt. Das bedeutet, daß sich Wartezeiten durch Refresh nur dann ergeben, wenn die CPU über längere Zeiträume hinweg ausschließlich diese RAM-Bank benutzt. Im ungünstigen Fall beträgt der Geschwindigkeitsverlust jedoch nur etwa 1 - 3%.

Der RAM-Block 2 liegt beim SAM-68K/256 normalerweise im Anschluß an Block 1 von der Adresse $20000 bis $3FFFF. Die obersten 32 KB davon werden als Video- und Grafik-RAM benutzt, der Rest steht wiederum dem Prozessor zur Verfügung. Soll SAM 68K mit externem RAM erweitert werden, so kann Block 2 in die obere Hälfte des 16 MB Adreßraumes verschoben werden oder im Anschluss an eine oder zwei 256 KB RAM-Erweiterungskarten gelegt werden (siehe Kapitel

7.9). Das ist wichtig, denn praktisch alle Betriebssysteme erwarten einen von Null an kontinuierlich durchgehenden RAM-Bereich. Der RAM Block 2 ist als Quasi Dual-Port RAM organisiert. Hierzu haben Prozessor und Videocontroller Zugriff. Wie schon erwähnt, kann der Prozessor über den Teil dieses Blockes frei verfügen, der nicht als Bildseite benötigt wird. Die Zugriffsdauer auf die RAM-Bank 2 hängt von der verwendeten Video-Auflösung (512x256 oder 512x512 Punkte wirken in dieser Beziehung gleich, es sind jedoch u.U. auch andere Auflösungen möglich) und von der Phasenlage zwischen Prozessor- und Videozugriff ab. In jedem Fall hat jedoch der Videocontroller beim Zugriff Vorrang vor dem Prozessor, um ein flimmerfreies Bild zu erreichen. Kommt es also darauf an, daß Programme optimal schnell ablaufen sollen, so sollten diese möglichst in der unteren RAM-Bank oder in externem RAM liegen. Der Geschwindigkeitsgewinn beträgt dadurch bis zu 50%.

Beim SAM-68K/640 liegt der RAM-Block 2 normalerweise fest von der Adresse $F00000 bis $F1FFFF.

Anmerkung:
In welcher Weise der RAM-Block 2 tatsächlich genutzt wird, ist ausschließlich durch die Software und die Systemumgebung bestimmt. Einerseits kann er durch entsprechende Programmierung des 6845 Controllers nur fuer Video-Output genutzt werden (z.B. Speicherung und schnelle Umschaltung zwischen mehreren Bildern). Andererseits kann dieser Speicher auch ausschliesslich als "Prozessor-RAM" benutzt werden, wenn beispielsweise der Rechner über ein Terminal oder eine zusätzliche Videokarte bedient wird.

Der EPROM-Bereich liegt ganz am oberen Ende des Adreßraums von $FE0000 bis $FEFFFF. Es sind zwei Stecksockel vorhanden (jeweils upper und lower Datenbyte), die mit Bausteinen der Typen 2732, 2764, 27128 oder 27256 bestückt werden können. Der Prozessorzugriff erfolgt im EPROM mit einem Wartezyklus; es müssen deshalb Bausteine mit einer Zugriffszeit von kleiner 300 nsec verwendet werden.

## 6.4 Adressdekodierung und Steuersignale

Dekodiert wird der gesamte Adreßbereich des Prozessors (16 MB) durch PAL F. Dieses PAL erzeugt die Chip-Select Signale für beide RAM-Bänke und die EPROMs, sowie ein Freigabesignal für die weitere Dekodierung der Ein/Ausgabe-Bausteine. Die Chip-Selects der peripheren Bausteine werden von einem TTL-Decoder erzeugt (LS 138).

PAL J ist zuständig für die Erzeugung des BUS TIMEOUT Signals. Erfolgt innerhalb von 8 usec nach Einleitung eines Zyklus durch den Prozessor keine Antwort (DTACK oder VPA Signal), so wird von PAL J der Zyklus mit einem Fehlersignal (BUS ERROR) abgebrochen. PAL J ist außerdem für die Einblendung des RESET Vektors nach dem

Rücksetzen des Prozessors verantwortlich. Dieser Vektor (Initial SSP und PC) muß in den ersten acht Bytes der EPROMs stehen. Die ersten 12 Speicherzugriffe nach einem RESET werden auf das EPROM umgelenkt!

PAL J erzeugt weiterhin ein VPA-Signal bei Zugriff auf die Peripheriebausteine auf der Karte. Dadurch erfolgen Zugriffe auf diese Bausteine synchron zum E-Takt des Prozessors.

## 6.5 Speicherbelegung

Die Speicherbelegung entnehmen Sie bitte der Tabelle 7.1 und den Ausführungen in Kapitel 6.3 "Speicher".

## 6.6 Ein/Ausgabe-Bausteine

Für parallele Ein/Ausgabe steht eine VIA 6522 zur Verfügung, deren Ein/Ausgänge auf den 50pol. Pfostenstecker für Flachkabelanschluß geführt sind (Tabelle 7.3). Dieser Baustein enthält darüber hinaus zwei 16 Bit Timer und ein Schieberegister. Näheres hierzu siehe Datenblatt.

Eine PIA 6821 dient zum Anschluß der parallelen Tastatur und zur Abfrage des Jumperfeldes J6 für die Einstellung der Systemparameter.

Die seriellen Schnittstellen von SAM 68K werden durch einen MPSC-Baustein (Multi Protokoll Serial Controller) unterstützt. Seine zwei unabhängigen Kanäle sind als asynchrone (z.B. V.24) oder synchrone Kanäle mit Übertragungsraten bis zu 800KB konfigurierbar.

Die Interrupt-Ausgänge der Peripheriebausteine sind zusammen auf die Interruptebene 3 gelegt.

## 6.7 Interrupts

Es sind sieben priorisierte Interruptleitungen vorhanden (I1 - I7), die über einen Encoder (74 LS 148) mit den drei Interrupt-Eingängen des Prozessors verbunden sind. I7 ist die Interruptleitung mit der höchsten Priorität (NMI), I1 die mit der niedrigsten Priorität. Die Ebenen I1, I2, I4, I5 sind frei verfügbar. Die auf der Platine befindlichen Interruptquellen (Peripheriebausteine, BREAK-Taste) benützen die Ebenen I3, I6, I7.

Pal Q dekodiert die Interrupt-Signale von den Peripheriebausteinen und beantwortet Interrupt Acknowledge Zyklen vom Prozessor so, dass ein von einem auf der Karte befindlichen Peripheriebaustein erzeugter Interrupt als Autovector Interrupt behandelt wird. An den Bus angeschlossene Interruptquellen koennen im Unterschied zu frueheren Versionen (2 - 4) von SAM 68k auf allen Ebenen auch User

Vector Interrupts verwenden. Bei einem Konflikt zwischen Autovector und User Vector Interrupts auf der gleichen Ebene hat IMMER (!) der Autovector Interrupt Vorrang. Zur Entwicklung eigener Zusatzbaugruppen mit User Vector Interrupts fordern Sie bitte bei kws genauere Unterlagen an.

Ein durch das Drücken der BREAK-Taste erzeugter Interrupt wird ebenfalls als Autovektor-Interrupt behandelt. Es ist auch möglich, externe Interruptquellen über den I7-Autovektor zu bedienen, dazu muß allerdings die BREAK-Routine durch Ändern dieses Vektors außer Funktion gesetzt werden und es sollte sichergestellt sein, daß das BREAK Monoflop nicht getriggert werden kann.

## 6.8 Bus Interface

Das Bus-Interface ermöglicht es, externe Baugruppen (RAM, Harddisk-Controller, A/D- D/A-Wandler, Grafikprozessoren u.a.), aus der EB 68000 Reihe an SAM 68K anzuschließen.

Das Bus Interface besteht aus sechs Treiberbausteinen (74 LS 245) und einigen Gattern. Sämtliche Busleitungen sind über die beiden 64pol. Stecker VG1 und VG2 nach außen geführt (Belegung siehe Tabelle 7.5 und 7.6).

Normalerweise sind die Puffer für die Adreßleitungen von SAM 68K zum Bus hin geschaltet. Steuerleitungen, wie AS, UDS, LDS, VMA, R/W und FC0 - FC2, werden wie Adreßleitungen behandelt. Die Richtung der Datenbuspuffer hängt von R/W und davon ab, ob ein Bauteil auf der Platine adressiert ist oder nicht.

RESET ist ein Ausgang und kann von außen nicht aktiviert werden.

Für DMA- und Multiprozessoranwendungen fordern Sie bitte nähere Informationen an.

## 6.9 Timeout

Da der Prozessor normalerweise eine Rückmeldung für den Abschluß eines Buszyklus benötigt, würde ein Zugriff auf eine nicht belegte Speicheradresse ohne besondere Maßnahmen einen unendlichen langen Buszyklus zur Folge haben. Um dies zu verhindern, wird ein Buszyklus nach 8 usec mit einem Fehlersignal (BUS ERROR) abgebrochen.

WICHTIG:

Beim Anschluß externer Baugruppen an SAM 68K sollte darauf geachtet werden, daß diese auf einen Zugriff hin innerhalb von 6 usec antworten.

## 6.10 Floppy-Disk-Controller

Als Floppy-Disk-Controller wird ein NEC 765 Baustein und ein

hochintegrierter Datenseparator verwendet. Diese beiden Bausteine übernehmen zusammen mit den Treiber-ICs die gesamte Steuerung der Laufwerke. Es können bis zu vier 5 1/4- oder 8 Zoll-Laufwerke angeschlossen werden.

Der 765-Controller zeichnet sich gegenüber anderen gebräuchlichen Floppy-Controller-Bausteinen durch eine hohe eigene Intelligenz aus. Seine größten Vorteile zeigt der Controller, wenn er im DMA-Mode betrieben wird. Um diesen Modus benutzen zu können, wurden verschiedene Möglichkeiten vorgesehen, das Datenregister des 765 anzusprechen:

1) ganz normal, ohne weitere Funktionen
2) gleichzeitig wird DMA-Acknowledge (DACK) aktiv gesetzt
3) gleichzeitig wird DMA-Acknoledge (DACK) und Terminal-Count (TC) aktiv gesetzt

Der Ausgang DMA-Request (DRQ) des 765 ist auf Port A Bit Nr. 7 (PA7) der PIA 6821 gelegt und kann von dort abgefragt werden.

Durch Ansprechen des Datenregisters nach Punkt 2) und 3) mit Unterstützung der PIA ist es möglich eine DMA-Betriebsart zu simulieren. Weiterhin kann durch "Low"-Setzen des Ausgangs CA2 der PIA 6821 erreicht werden, daß Interrupts des Floppy-Controllers gesperrt werden.

Da der 765 bei jeder Positionierung ein Head-Unload Signal gibt, kann es bei Drives älterer Bauart zu unerwünschten Nebengeräuschen durch häufiges Kopfladen kommen. Um dies zu unterbinden, kann der Headload völlig unabhängig vom Controller über den PIA-Pin PA0 gesteuert werden (BR19).

5" Drives benötigen in der Regel eine Motor ein/aus Steuerung. Diese kann durch Pin PA1 der PIA bedient werden.

## 6.11 Video-Controller

Der Videoteil des SAM 68K wird durch einen 6845 Controller gesteuert. Er erzeugt alle nötigen Signale für die Synchronisation des Video-Monitors und bestimmt das Format des auf dem Monitor abgebildeten Bildes. Der 6845 kann auf den gesamten zweiten RAM-Block zugreifen. Er hat hierbei Vorrang vor dem Prozessor.

Normalerweise wird der 6845 so programmiert, daß sich eine Bildschirmauflösung von 512 Punkten in der Horizontalen und 256 Zeilen in der Vertikalen ergibt. Jedem Punkt auf dem Bildschirm entspricht ein Bit eines Bytes desjenigen Speicherbereiches, der auf dem Bildschirm abgebildet wird. Ist ein Bit auf log. "1" gesetzt, so wird der zugehörige Bildpunkt "eingeschaltet", andernfalls bleibt er dunkel. Die Bits bzw. Bytes werden mit aufsteigenden Adressen von links nach rechts und von oben nach unten auf dem Bildschirm abgebildet; d.h. die niedrigste Adresse der Bildseite liegt links oben im Bild, Bit Nr.7 dieses Bytes

sitzt dabei ganz in der Ecke. Die Bildwechselfrequenz bei dieser Auflösung beträgt 50 Hz (non-interlaced mode).

Durch eine andere Initialisierung des 6845-Contollers kann erreicht werden, daß die Auflösung in der Vertikalen auf 512 Zeilen erhöht wird. Die lineare Zuordnung von Bits und Bytes im Speicher und Bildpunkten auf dem Monitor bleibt dabei erhalten. Es wird hierbei jedoch im interlaced-Mode (Zeilensprung) gearbeitet mit einer Bildfrequenz von 25 Hz, sodaß in den meisten Fällen ein Video-Monitor mit einer lang nachleuchtenden Bildröhre (P39 Phosphor) erforderlich ist.

Bei 768 x 480 Bildpunkten gilt die lineare Adresszuordnung nicht mehr; Genaueres dazu siehe Kap. 8.2.

Der BAS-Ausgang des Videoteils enspricht den üblichen Normen, er liefert ca. 1.5V (Spitze/Spitze) an 100 Ohm Eingängen. Bei Versionen mit TTL-Ausgang (getrennte Synchronsignale) betraegt der Pegel des Videosignals ebenfalls ca. 1.5V an 100 Ohm; die Synchronsignale sind positive TTL-Signale.

# 7 Tabellen und Schaltpläne

## 7.1 Speicherbelegung und Adressen der Ein/Ausgabe-Bausteine

Speicherbelegung

| | SAM-68K/256 von bis KB | SAM-68K/640 von bis KB |
|---|---|---|
| RAM-Block 1 | 000000 - 01FFFF 128 | 000000 - 07FFFF 512 |
| RAM-Block 2 | | |
| normal a) | 020000 - 03FFFF 128 | F00000 - F1FFFF 128 |
| wahlweise b) | F00000 - F1FFFF 128 | 080000 - 09FFFF 128 |
| oder c) | 060000 - 07FFFF 128 | 0C0000 - 0DFFFF 128 |
| oder d) | 0A0000 - 0BFFFF 128 | 100000 - 11FFFF 128 |
| siehe 1) | | |
| frei fuer Erweiterungen 2) | Ende RAM - EFFFFF | Ende RAM - EFFFFF |
| EPROM | FE0000 - FEFFFF 64 | FE0000 - FEFFFF 64 |
| I/O-Baugruppen 3) | FF0000 - FF7FFF 32 | FF0000 - FF7FFF 32 |
| Peripherie-bausteine | FF8800 - FF8FFF | FF8800 - FF8FFF |
| reserviert fuer System Controller | FFF000 - FFF7FF | FFF000 - FFF7FF |

1) Die Lage des RAM-Blocks 2 wird durch die Jumper J6 6-13 und J6 6-14 bestimmt. Siehe hierzu auch die Beschreibung der Jumper und das Kapitel 6.3 "Speicher". Die Möglichkeiten b), c) und d) werden beim SAM-68K/640 von der Monitorsoftware nicht unterstützt!

2) innerhalb dieses Adreßbereiches sind die Buspuffer so

gerichtet, daß Bausteine außerhalb der Karte angesprochen werden können.

3) In diesem Bereich liegen die meisten der von kws erhaeltlichen Zusatzbaugruppen. Siehe auch 2).

## Adressen der Ein/Ausgabe-Bausteine

### VIA 6522

| | |
|---|---|
| FF8801 | Port B Data |
| FF8803 | Port A Data |
| FF8805 | Port B Direction |
| FF8807 | Port A Direction |
| FF8809 | Timer 1 Counter/Latch Low |
| FF880B | Timer 1 Counter High |
| FF880D | Timer 1 Latch Low |
| FF880F | Timer 1 Latch High |
| FF8811 | Timer 2 Counter/Latch Low |
| FF8813 | Timer 2 Counter High |
| FF8815 | Shift Register |
| FF8817 | Auxiliary Control Register |
| FF8819 | Peripheral Control Register |
| FF881B | Interrupt Flag Register |
| FF881D | Interrupt Enable Register |
| FF881F | Port A Data |

### PIA 6821

| | |
|---|---|
| FF8901 | Port A Data |
| FF8903 | Port A Control |
| FF8905 | Port B Data |
| FF8907 | Port B Control |

### CRT-Controller 6845

| | |
|---|---|
| FF8A01 | CRT Adress Register |
| FF8A03 | CRT Register File |

### Multi-Protokoll Serial Controller (7201/8274)

| | |
|---|---|
| FF8C01 | Data Register A |
| FF8C03 | Data Register B |
| FF8C05 | Control Register A |
| FF8C07 | Control Register B |

Floppy-Disk-Controller 765

| | |
|---|---|
| FF8D01 | FDC Status Register |
| FF8D03 | FDC Daten Register |
| FF8E03 | FDC Daten Register mit DMA-Service und Terminal Count |
| FF8E43 | FDC Datenregister mit DMA-Service |

## 7.2 Belegung des Steckverbinder PF1 für serielle Ein/Ausgabe

| PIN Nr. | belegt mit | Bemerkungen |
|---|---|---|
| 1 | Masse | |
| 2 | Masse | |
| 3 | TxClk1 | |
| 4 | RxClk1 | Clock Signale, für SDLC, HDLC |
| 5 | Masse | etc. Betrieb. Bei Betrieb als |
| 6 | Masse | V.24 Schnittstellen muessen |
| 7 | TxClk2 | Pins 3-4 und 7-8 per Jumper |
| 8 | RxClk2 | kurzgeschlossen werden |
| 9 | Masse | |
| 10 | Masse | |
| 11 | CD1, | |
| 12 | DTR1 | |
| 13 | SYNC1 | |
| 14 | RTS1* | |
| 15 | CTS1* | |
| 16 | RxD1* | |
| 17 | TxD1* | Ein/Ausgänge, die direkt am |
| 18 | CD2 | MPSC angeschlossen sind |
| 19 | DTR2* | |
| 20 | SYNC2*/RTS2* | |
| 21 | CTS2* | |
| 22 | RxD2* | |
| 23 | TxD2* | |
| 24 | + 5V | |
| 25 | +12V | |
| 26 | -12V | |

Die Ziffern 1 und 2 der Signalbezeichnungen beziehen sich auf die zwei unterschiedlichen Kanäle 1 und 2. Mit einem Stern gekennzeichnete Signale sind low-aktiv.

Siehe hierzu auch Schaltplan MPSC-Baustein.

Soll der MPSC die V.24-Schnittstellen unterstützen, dann müssen die PINs 3-4 bzw. 7-8 auf diesem Stecker mit einem Jumper kurzgeschlossen werden.

## 7.3 Belegung des Steckverbinders PF2

| PIN Nr. | belegt mit | Funktion |
|---|---|---|
| 1 | + 12V | |
| 2 | - 12V | |
| 3 | TxD1 V.24 | |
| 4 | RxD1 V.24 | V.24- |
| 5 | RTS1 V.24 | |
| 6 | CTS1 V.24 | Schnittstellen |
| 7 | Masse | |
| 8 | TxD2 V.24 | |
| 9 | RxD2 V.24 | |
| 10 | RTS2 V.24 | |
| 11 | CTS2 V.24 | |
| 12 | Masse | |
| 13 | Masse | |
| 14 | Masse | |
| 15 | frei | |
| 16 | frei | |
| 17 | PA0 | WEN* 1) |
| 18 | PB0 | Bit 0 2) |
| 19 | PA1 | CIP* 1) |
| 20 | PB1 | Bit 1 2) |
| 21 | PA2 | RDA* 1) |
| 22 | PB2 | Bit 2 2) |
| 23 | PA3 | RDC 1) |
| 24 | PB3 | Bit 3 2) |
| 25 | PA4 | WDA* 1) |
| 26 | PB4 | Bit 4 2) |
| 27 | PA5 | WCD* 1) |
| 28 | PB5 | Bit 5 2) |
| 29 | PA6 | FWD* 1) |
| 30 | PB6 | Bit 6 2) |
| 31 | PA7 | REV* 1) |
| 32 | PB7 | BUSY 2) |
| 33 | CA1 | WCLK 1) |
| 34 | CB1 | reserviert 2) |
| 35 | CA2 | OK* 1) |
| 36 | CB2 | Data Strobe 2) |
| 37 | + 5V | |
| 38 | + 5V | |
| 39 | Bit 0 | |
| 40 | Bit 1 | |
| 41 | Bit 2 | |
| 42 | Bit 3 | |
| 43 | Bit 4 | Tastaturanschluß |
| 44 | Bit 5 | |
| 45 | Bit 6 | |
| 46 | Bit 7 | |
| 47 | Strobe | |
| 48 | CB2 | |
| 49 | Masse | |
| 50 | Masse | |

1) Steuerleitungen für MiniDCR, 2) Leitungen für Centronics
Alle E/A-Leitungen sind für den Benutzer voll verfügbar, solange diese Treiber nicht angesprochen werden. Erst dann werden sie initialisiert.

## 7.4 Pin Belegung der Steckverbinder PF3 und PF4 für Floppy-Disk Anschluß

| Funktion | Pin Nr. 8"(PF3) | Pin Nr. 5"(PF4) | |
|---|---|---|---|
| Headload HDL | 18 | 2 | |
| Drive Sel. 0 | 26 | 10 | |
| Drive Sel. 1 | 28 | 12 | |
| Drive Sel. 2 | 30 | 14 | |
| Drive Sel. 3 | 32 | frei | |
| 2 Sided | 10 | frei | * |
| Write Prot. | 44 | 28 | |
| Track 0 | 42 | 26 | |
| Fault | frei | frei | |
| Fault Reset | frei | frei | |
| Step | 36 | 20 | |
| Low Current | 2 | frei | |
| Direction | 34 | 18 | |
| Index | 20 | 8 | |
| Ready | 22 | 6 | |
| Side select | 14 | 32 | |
| Write Gate | 40 | 24 | |
| Write Data | 38 | 22 | |
| Read Data | 46 | 30 | |
| Motor on | frei | 16 | |
| Alt. Ready | frei | 34 | |

Alle ungeraden Pins auf Masse

*: Fuer den Betrieb von 5 1/4-Zoll Laufwerken mit zweiseitig beschriebenen Disketten MUSS BR17 kurzgeschlossen sein. Siehe 7.9.

## 7.5 Belegung des Bussteckers VG 1 (beim Speicher)

| Beschreibung | Bez. | Pin # | | Bez. | Beschreibung |
|---|---|---|---|---|---|
| | Masse | 1a | 1c | Masse | |
| | Masse | 2a | 2c | Masse | |
| | +12V | 3a | 3c | -12V | |
| | | 4a | 4c | BD0 | |
| | | 5a | 5c | BD1 | |
| | | 6a | 6c | BD2 | |
| reserviert | | 7a | 7c | BD3 | Datenleitungen |
| | | 8a | 8c | BD4 | |
| | | 9a | 9c | BD5 | |
| | | 10a | 10c | BD6 | |
| | | 11a | 11c | BD7 | |
| reserviert | | 12a | 12c | | reserviert |
| | | 13a | 13c | BD8 | |
| | | 14a | 14c | BD9 | |
| | | 15a | 15c | BD10 | |
| reserviert | | 16a | 16c | BD11 | Datenleitungen |
| | | 17a | 17c | BD12 | |
| | | 18a | 18c | BD13 | |
| | | 19a | 19c | BD14 | |
| | | 20a | 20c | BD15 | |
| | | 21a | 21c | | |
| | | 22a | 22c | | |
| | | 23a | 23c | BA1 | |
| | | 24a | 24c | BA2 | |
| | | 25a | 25c | BA3 | |
| reserviert | | 26a | 26c | BA4 | Adress- |
| | | 27a | 27c | BA5 | leitungen |
| | | 28a | 28c | BA6 | |
| | | 29a | 29c | BA7 | |
| | | 30a | 30c | BA8 | |
| | +5V | 31a | 31c | +5V | |
| | +5V | 32a | 32c | +5V | |

## 7.6 Belegung des Bussteckers VG 2 (beim Prozessor)

| Beschreibung | Bez. | Pin # | | Bez. | Beschreibung |
|---|---|---|---|---|---|
| | | 1a | 1c | | |
| | +5V | 2a | 2c | +5V | |
| | +5V | 3a | 3c | +5V | |
| reserviert | | 4a | 4c | | |
| Bus Gr. Ackn. | BGACK* | 5a | 5c | | reserviert |
| | | 6a | 6c | | |
| reserviert | | 7a | 7c | | |
| | | 8a | 8c | | |
| Int. Control | IVP | 9a | 9c | DTACK* | Data Acknowledge |
| Bus Grant | BG* | 10a | 10c | | reserviert |
| - | BA23 | 11a | 11c | BA22 | - |
| Adress- | BA21 | 12a | 12c | BA20 | Adress- |
| leitungen | BA19 | 13a | 13c | BA18 | leitungen |
| - | BA17 | 14a | 14c | BA16 | - |
| reserviert | | 15a | 15c | | reserviert |
| reserviert | | 16a | 16c | | reserviert |
| - | BA15 | 17a | 17c | BERR* | Bus Error |
| Adress- | BA13 | 18a | 18c | BA14 | - Adress- |
| leitungen | BA11 | 19a | 19c | BA12 | leitungen |
| - | BA9 | 20a | 20c | BA10 | - |
| - | I5* | 21a | 21c | BAS* | Address Strobe |
| Interrupt | I7* | 22a | 22c | I4* | - |
| | I2* | 23a | 23c | I6* | Interrupt |
| - | I3* | 24a | 24c | I1* | - |
| E - Clock | BE* | 25a | 25c | BUDS* | Upper Data Strobe |
| - | BFC0 | 26a | 26c | BLDS* | Lower Data Strobe |
| Function | BFC1 | 27a | 27c | BR/W | Read/Write |
| Codes - | BFC2 | 28a | 28c | BVMA* | Valid Memory Addr. |
| | BHALTIN* | 29a | 29c | BCLC | Clock |
| Bus Request | BR* | 30a | 30c | BHALTOUT* | |
| Valid Per. Addr. | VPA* | 31a | 31c | BRES* | Reset |
| | Masse | 32a | 32c | Masse | |
| | Masse | | | Masse | |

## 7.7 Einstellung der Baudrate der V.24 Schnittstellen

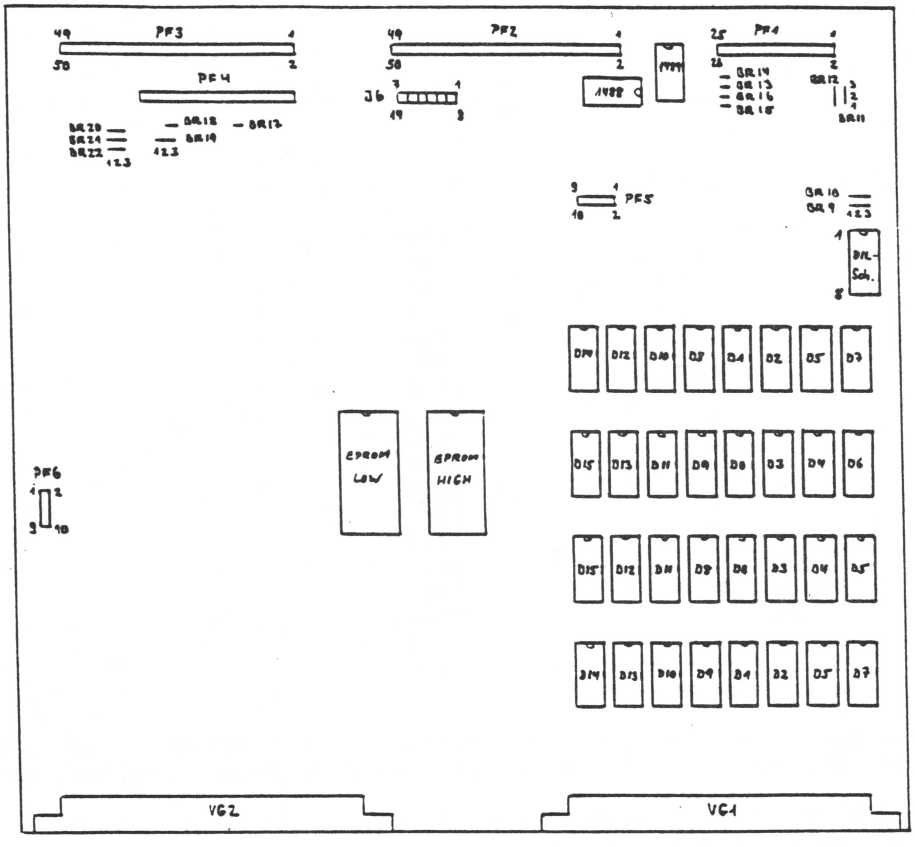
Die Übertragungsrate der V.24 Schnittstellen kann für beide Kanäle getrennt eingestellt werden. Für Kanal 1 sind die Schalter 1 bis 4, für Kanal Nr. 2 die Schalter 5 bis 8 des 8-fach DIL-Schalters zuständig. Die Angaben setzen den standardmäßig programmierten Teilungfaktor Clock/16 voraus.

| DIL-Schalter Nr. | | |
|---|---|---|
| 1 2 3 4 | für V.24 Nr. 1 | |
| 5 6 7 8 | für V.24 Nr. 2 | |
| | Baudrate | |
| 1 1 1 1 | 50 | |
| 1 1 1 0 | 75 | |
| 1 1 0 1 | 110 | |
| 1 1 0 0 | 134.5 | |
| 1 0 1 1 | 150 | |
| 1 0 1 0 | 300 | |
| 1 0 0 1 | 600 | |
| 1 0 0 0 | 1200 | |
| 0 1 1 1 | 1800 | |
| 0 1 1 0 | 2000 | |
| 0 1 0 1 | 2400 | |
| 0 1 0 0 | 3680 | |
| 0 0 1 1 | 4800 | |
| 0 0 1 0 | 7200 | |
| 0 0 0 1 | 9600 | |
| 0 0 0 0 | 19200 * | |

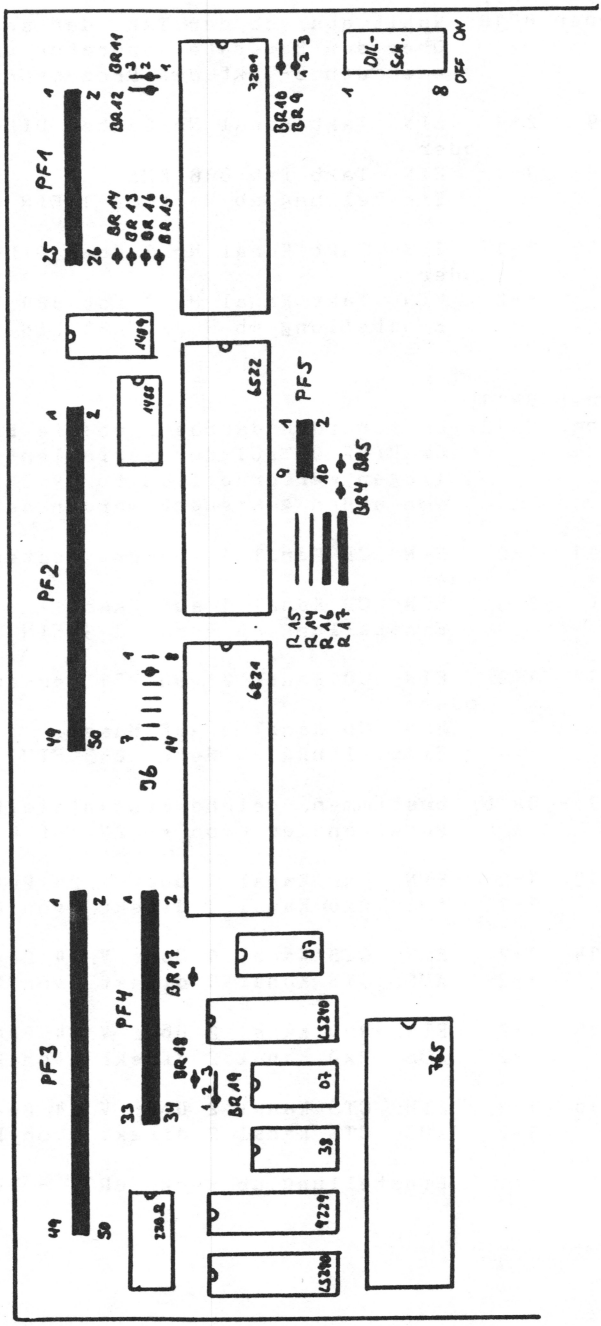
* Dieser Wert stimmt nicht genau. In Wirklichkeit sind es 19800 Baud. Dies ist jedoch für die meisten Anwendungen unerheblich.

In der Tabelle bedeutet "1" Schalter in Position "on", "0" Schalter in Position "off".

## 7.8 Jumperplan SAM 68K



Bilder: Genaue Bezeichnung und Lage der Jumper
Eingezeichnet ist jeweils die Jumperstellung, die ab Werk eingestellt wird.



## 7.9 Bedeutung der Jumper

Jumper BR9,
Jumper BR10: Wählt aus, ob der Takt der seriellen Kanäle über den Baudrate-Generator und den DIL-Schalter, oder den E-Takt des Prozessors generiert wird.

BR9 2-1 EIN: Takt Kanal Nr.2 über DIL-Schalter
oder
3-2 EIN: Takt ist 800 KHz
Einstellung ab Werk: 2-1 EIN

BR10 2-1 EIN: Takt Kanal Nr.1 über DIL-Schalter
oder
3-2 EIN: Takt Kanal Nr.1 ist 800 KHz
Einstellung ab Werk: 2-1 EIN

Jumper BR11,
Jumper BR12: Entscheiden darüber, ob die Eingangssignale DATA CARRIER DETECT der seriellen Schnittstellen auf Masse liegen (erforderlich für V.24 Betrieb) oder von von außen gesteuert werden sollen.

BR11 1-2 EIN: CD Kanal 1 von PF1 gesteuert
oder
2-3 EIN: CD Kanal 1 auf Masse
Einstellung ab Werk: 2-3 EIN

BR12 1-2 EIN: CD Kanal 2 von PF1 gesteuert
oder
2-3 EIN: CD Kanal 2 auf Masse
Einstellung ab Werk: 2-3 EIN

BR13 - BR16: bestimmen, welche Eingangsleitungen über V.24 Pegelwandler (von +-12V auf 0-5V) laufen sollen.

BR13 1-2 EIN: RxD Kanal 1 über V.24 Pegelwandler
1-2 AUS: RxD Kanal 1 direkt (von PF1)

BR14 1-2 EIN: CTS Kanal 1 über V.24 Pegelwandler
1-2 AUS: CTS Kanal 1 direkt (von PF1)

BR15 1-2 EIN: RxD Kanal 2 über V.24 Pegelwandler
1-2 AUS: RxD Kanal 2 direkt (von PF1)

BR16 1-2 EIN: CTS Kanal 2 über V.24 Pegelwandler
1-2 AUS: CTS Kanal 2 direkt (von PF1)

Einstellung ab Werk: BR13 - BR16 EIN

Pfostensteckverbinder PF1: Betrieb mit V.24 Schnittstellen

PF1 3-4 EIN und
7-8 EIN: verbindet jeweils TxClk und RxClk der beiden seriellen Kanäle miteinander. Diese beiden Jumper muessen gesetzt sein, wenn mit V.24 Schnittstellen gearbeitet wird.
Einstellung ab Werk: 3-4 und 7-8 EIN

Jumper J6: dient zum Anschluß eines Lichtgriffels und zum Festlegen der Systemparameter

J6 8 + 5V Versorgung für Lichtgriffel
ACHTUNG: J6 Pins 2 - 7 liegen auf Masse !

2-9 EIN: Betrieb ohne Lichtgriffel
2-9 sonst: Anschluß des Lichtgriffels (siehe Bild)
Einstellung ab Werk: 2-9 EIN

3-10 EIN: Automatisches Booten des Betriebssytems nach Einschalten des Rechners
3-10 AUS: Normalbetrieb
Einstellung ab Werk: 3-10 AUS

4-11 EIN: Betrieb mit 8" Floppy
4-11 AUS: Betrieb mit 5" Floppy
Einstellung ab Werk: 4-11 AUS

5-12 EIN: Konsole ist Terminal an der V.24 Nr. 1
5-12 AUS: Konsole ist Tastatur/Bildschirm
Einstellung ab Werk: 5-12 AUS

6-13 und 7-14 bestimmen die Lage des RAM-Blocks 2 (VIDEO-RAM) nach folgendem Schema (beachten Sie die Unterschiede bei SAM-68K/256 (256 KB-Version) und SAM-68K/640 (640 KB Version!)

| Jumper 6-13 | Jumper 7-14 | bei SAM-68K/256 RAM-Block 2 von | bis | bei SAM-68K/640 RAM-Block 2 von | bis |
|---|---|---|---|---|---|
| AUS | AUS | 02 0000 | 03 FFFF * | 08 0000 | 09 FFFF |
| EIN | AUS | F0 0000 | F1 FFFF | F0 0000 | F1 FFFF * |
| AUS | EIN | 06 0000 | 07 FFFF | 0C 0000 | 0D FFFF |
| EIN | EIN | 0A 0000 | 0B FFFF | 10 0000 | 11 FFFF |

Einstellung ab Werk: durch "*" markiert

Jumper BR17: Floppy-Disk-Betrieb. Two-sided.

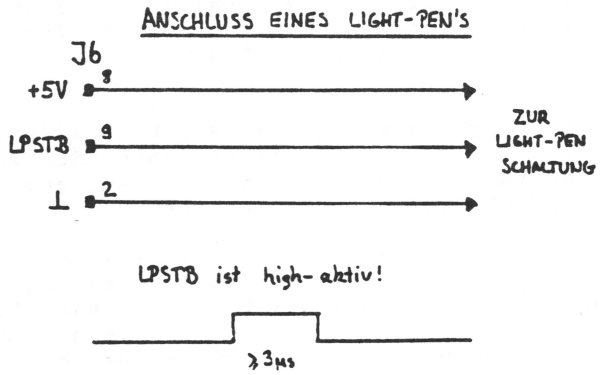
BR17 1-2 EIN: es werden zweiseitige Laufwerke verwendet
1-2 AUS: es werden einseitige Laufwerke verwendet
Einstellung ab Werk: 1-2 EIN
Wird nur bei 5" Laufwerken benötigt
Bei 8" Floppys muß J13 unbedingt auf "AUS" stehen!

Jumper BR18: Floppy-Disk-Betrieb. Drive Ready Signal.

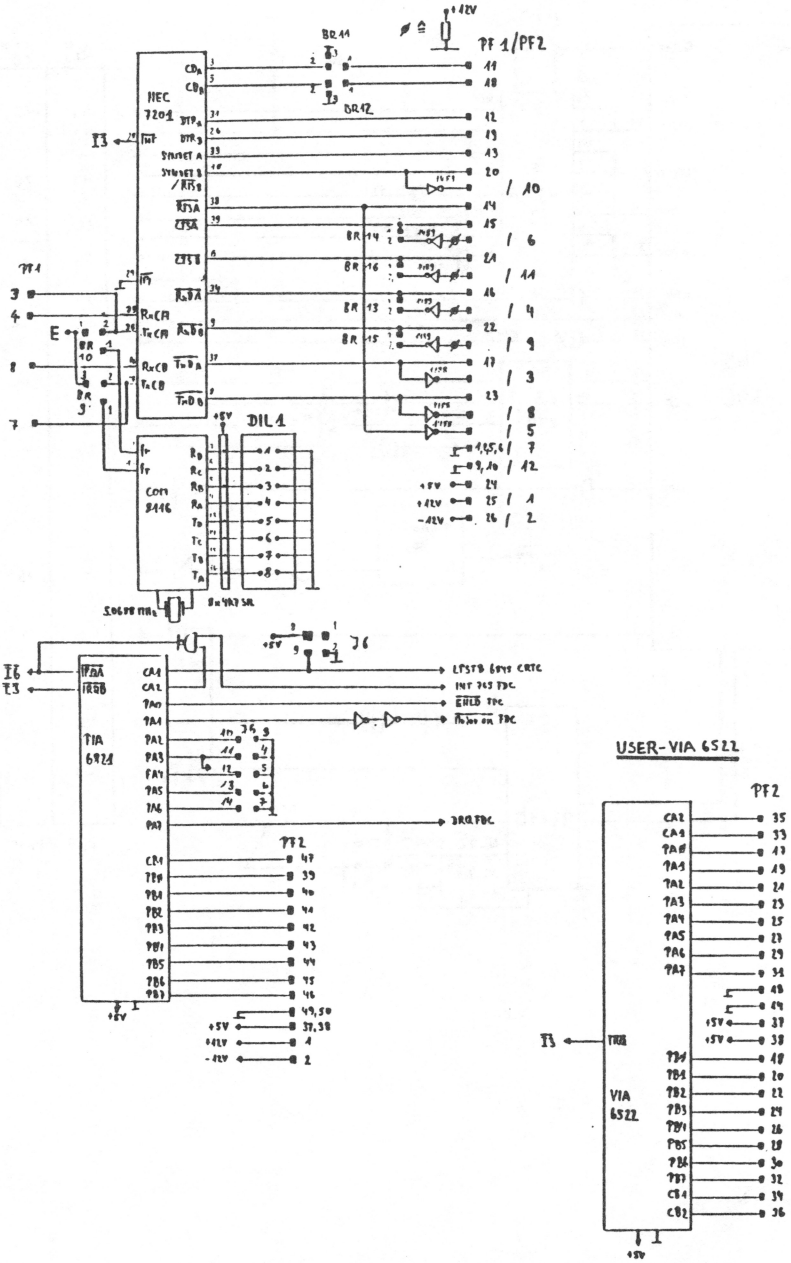
BR18 1-2 EIN: Alternate Ready Leitung in Betrieb
1-2 AUS: Alternate Ready Leitung außer Betrieb
Einstellung ab Werk: 1-2 EIN
Wird nur für 5" Laufwerke benötigt

Jumper BR19: Floppy-Disk-Betrieb. Headload/Motor-Steuerung

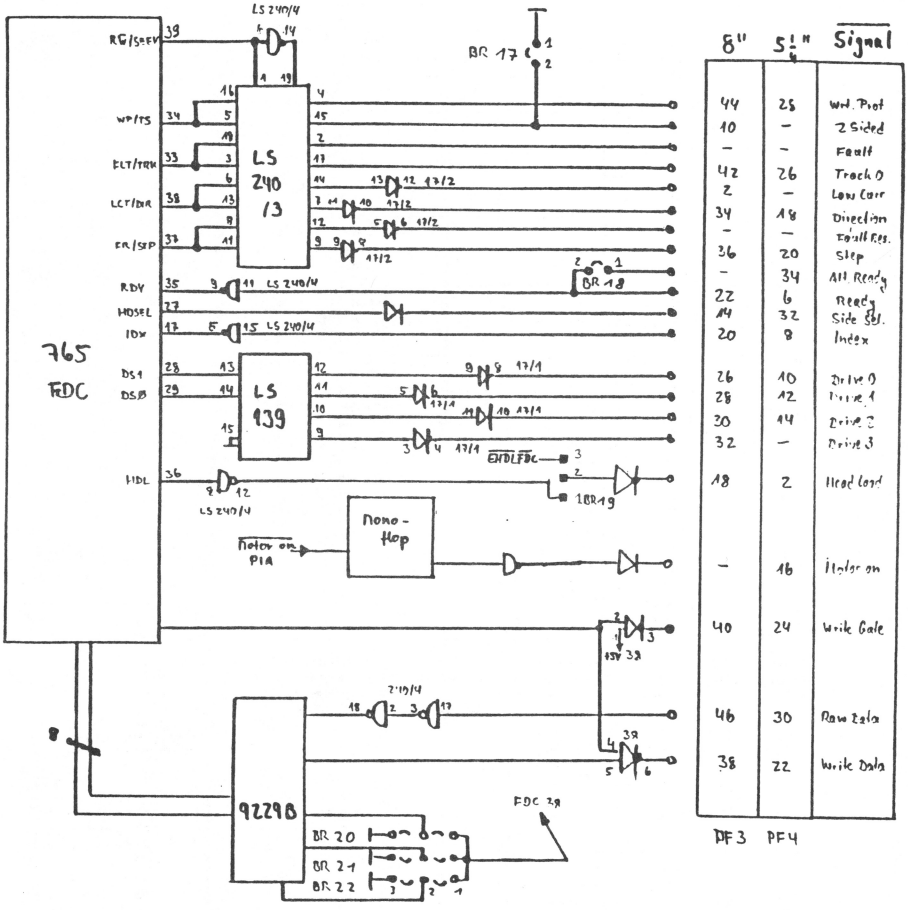
BR19 1-2 EIN: Headload direkt
oder
2-3 EIN: Headload von PIA 6821 PA0
Einstellung ab Werk 1-2 EIN



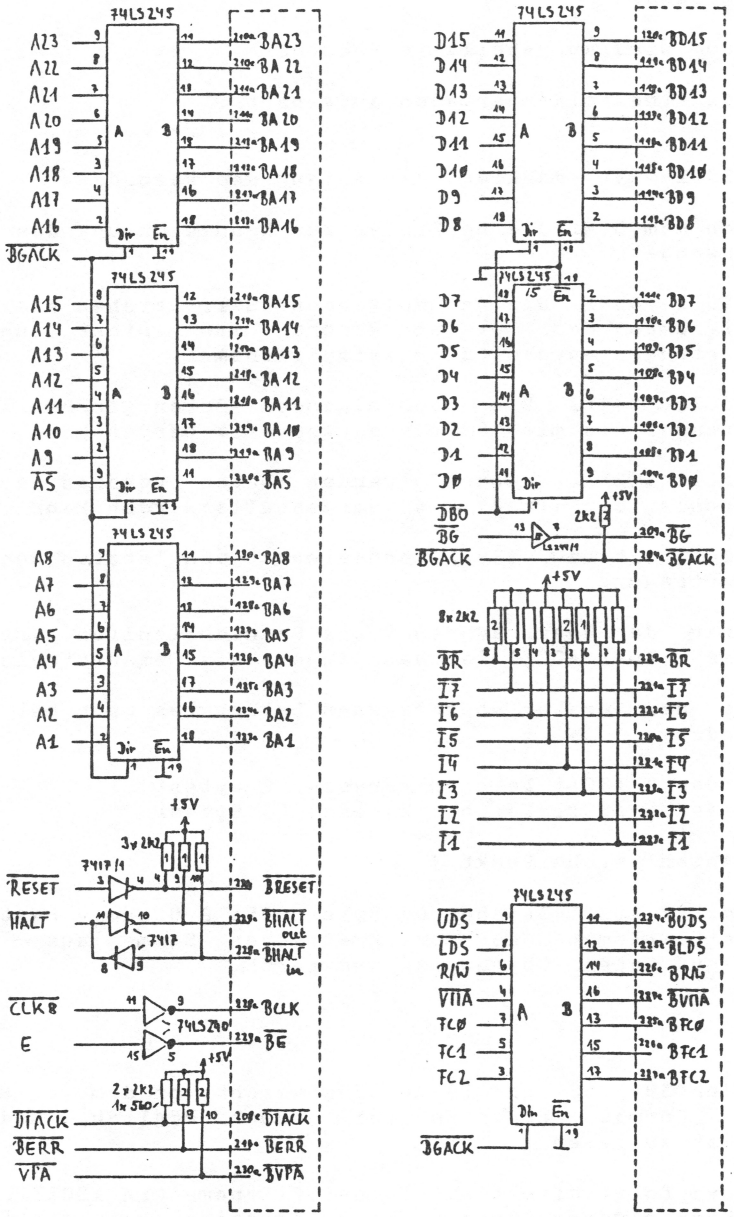
## 7.10 Schaltplan des Ein/Ausgabe-Teils



## 7.11 Schaltplan des Floppy-Disk-Interfaces



## 7.12 Schaltplan des Bus-Interfaces



## 7.13 Verwendetes Format für Binärcode (Motorola kompatibel)

Zur Übertragung wird jedes Byte in zwei 4-Bit-Hälften (Nibbles) zerlegt. Diese werden als Hexadezimalzahlen interpretiert und im ASCII-Code gesendet.

Der Datenstrom wird aufgeteilt in Records.

Diese Records sind folgendermaßen aufgebaut:

1.) S (ASCII $53), markiert den Anfang des Records

2.) Nach dem S wird eine Ziffer zur Kennzeichnung des Records verwendet:

- 0 (ASCII $30). Die nachfolgenden ASCII-Zeichen stellen den Filenamen dar. Dieser Record muß nicht unbedingt übertragen werden (dann kein Filename).
- 1 (ASCII $31). Die nachfolgenden Daten gehören zu einer Adresse, die mit 16 Bits dargestellt werden kann.
- 2 (ASCII $32). Die nachfolgenden Daten gehören zu einer Adresse, die mit 24 Bits dargestellt werden kann.
- 8,9 (ASCII $38 bzw. $39), kennzeichnet den letzten Record der Übertragung.

3.) Die Anzahl der übertragenen Bytes (einschließlich Adresse und Prüfsumme). Zwei ASCII-Zeichen, Angabe als Hexadezimalzahl.

4.) Adresse des ersten übertragenen Datenbytes (nur bei S1- und S2-Records)

S1-Records: 4 ASCII-Zeichen Adresse (2 Bytes)
S2-Records: 6 ASCII-Zeichen Adresse (3 Bytes)

5.) Daten. Anzahl siehe Punkt 3.

6.) Prüfsumme, 2 ASCII-Zeichen (1 Byte). Die Prüfsumme besteht aus dem 1er-Komplement der arithmetischen Summe aus Anzahl, Adresse und Daten, (byteweise summiert).

Anmerkungen:

Das Format der S0-, S8- und S9-Records weicht von dem von Motorola verwendeten Format ab. Es ist jedoch möglich, Code nach Motorolaformat zu lesen.

Beim S0-Record folgt direkt auf S0 der Filename (in ASCII), keine Anzahl der übertragenen Zeichen, keine Adressen, keine Prüfsumme.

S8- und S9-Records werden ohne Zusatz als Endekennung verwendet.

Records werden in der Regel durch <cr> und LF getrennt.

Beispiele:

S0PROGRAMM1

S11311000000FFFF0000FFFF0000FFFF0000FFFFE3

S214FF80000000050000FF80246000053260000 4FACF

S9

## 7.14 RAM - Belegung durch die Software

Stand: Monitor Version 1.0

| Von Adresse | bis Adresse | belegt durch |
|---|---|---|
| $00 0000 | $00 03FF | Vektortabelle |
| $00 0400 | $00 04DF | Monitor Stack |
| $00 04E0 | $00 06FF | Monitor Arbeitsbereich |
| $00 067E | Ram-Ende | frei |

Wichtig:
Benutzerprogramme sollten stets einen eigenen Stack benutzen!

Die genaue Lage der Monitorvariablen entnehmen Sie bitte dem Listing des Monitorprogramms.

Der vom Videocontroller auf dem Bildschirm abgebildete Speicherbereich beginnt normalerweise bei der Anfangsadresse des Videospeichers + $18000 (bei 512 x 256 oder 512 x 512) bzw. bei Anfangsadresse + $14C00 (bei 768 x 480).

Beispiele:

| | 512 x 256 | 512 x 512 | 768 x 480 |
|---|---|---|---|
| Start: | $038000 | $038000 | $034C00 |
| Ende: | $03BFFF | $03FFFF | $03FFFF |

oder

| | | | |
|---|---|---|---|
| Start: | $F18000 | $F18000 | $F14C00 |
| Ende: | $F1BFFF | $F1FFFF | $F1FFFF |

Die Lage der Videoseite ist abhaengig von der Stellung der Jumper

J6 6-13 und J6 7-14. Sie wird von den Videotreibern des Monitors automatisch erkannt bei SAM-68K/256, bei SAM68K/640 sollte der Videospeicher bei $F00000 beginnen (J6 6-13 EIN, 7-14 AUS). Fuer spezielle Anwendungen (z.B. Umschalten verschiedener Bilder) kann der im Bild erscheinende Speicherbereich innerhalb des Videospeichers verschoben werden. Dazu ist eine Aenderung der Startadresse in der Registertabelle fuer den Videocontroller 6845 notwendig.

## 7.15 Steuerzeichen der Videotreiber

Control Codes

| Contr. Zeich. | hex. Wert | Funktion |
|---|---|---|
| Control M | $0D | Carriage Return |
| Control J | $0A | Line Feed |
| Control H | $08 | Backspace |
| Control B | $02 | Cursor home Down |
| Control K | $0B | Cursor home Up |
| Control L | $0C | Cursor home/Clear screen |
| Control O | $0F | Cursor up |
| Control N | $0E | Cursor right |
| Control E | $05 | Erase till end of line |
| Control P | $10 | Erase till end of page |
| Control G | $07 | Piepser |

Escape Sequenzen

| Senden von | bewirkt |
|---|---|
| ESC A | US-ASCII Zeichensatz einschalten |
| ESC B | CURSOR einschalten |
| ESC C z s | CURSOR auf die angegebene Zeile und Spalte positionieren 1) |
| ESC D | Deutschen Zeichensatz einschalten |
| ESC E | CURSOR ausschalten |
| ESC F | invertierte Schrift einschalten |
| ESC G | invertierte Schrift ausschalten |

1) z und s bedeuten den Code für die gewünschte Zeile und Spalte. Zeilen zählen von oben nach unten von 0 bis 24, Spalten von links nach rechts von 0 bis 79. Der Code setzt sich zusammen aus der Nummer der Zeile oder Spalte plus $20.

Die von der Tastatur gesendeten Zeichen werden folgendermaßen umcodiert:

| gesendeter Code | weitergeleitet wird |
|---|---|
| $B8 | $04 |
| $B3 | $06 |
| $B1 | $7F |
| $BF | $02 |
| $0C | $1C |
| $09 | $0E |
| $B7 | $0B |
| $0B | $0F |
| $C0 | $05 |
| $B2 | $13 |
| $C2 | $2E |
| $B5 | $2B |
| $BA | $2D |
| $BD | $09 |
| $C3 | $0D |

Alle anderen Zeichen, die die Tastatur sendet, werden unverändert weitergeleitet.

## 7.16 Belegte Trap-Vektoren und Interrupt-Ebenen

| | |
|---|---|
| TRAP 0 | Abbruch eines Programms mit Registerdump |
| TRAP 1 | Sprungtabelle nach SUSY-Norm liefern |
| TRAP 2 | frei, jedoch durch SUSY-Betriebssystem belegt |
| TRAP 14 | Breakpoints |
| TRAP 15 | Ein/Ausgabe-TRAP |

| | |
|---|---|
| Interrupt-Ebene 3: | Ein/Ausgabe, Peripheriebausteine |
| Interrupt-Ebene 6: | Floppy-Disk-Controller |
| Interrupt-Ebene 7: | BREAK-Taste |

## 7.17 Belegung des Steckverbinders PF5 für den Anschluß eines Videomonitors

| PIN Nr. | belegt mit |
|---|---|
| 1 | Masse |
| 2 | VSync |
| 3 | +5V |
| 4 | Masse |
| 5 | +5V |
| 6 | HSync |
| 7 | +5V |
| 8 | Masse |
| 9 | +5V |
| 10 | Video (BAS) |

Sollen getrennte Bild- und Synchron-Signale entnommen werden, so müssen Drahtbrücken und Widerstände anders geschaltet werden:

BR5 und BR4 muessen entfernt werden und R17 und R15 mit jeweils 220 Ohm bestueckt werden.

Spezielle Monitore verlangen ein Synchronsignal, das Horizontal- und Vertikalsynchronimpulse gemischt enthaelt. Dazu muss BR4 entfernt und R15 mit 220 Ohm bestueckt werden.

Standardmäßig werden die Platinen so ausgeliefert, dass an PIN 8 und 10 das VIDEO-Signal(BAS) entnommen werden kann. Das BAS-Signal enthaelt Bildinhalt und Synchronsignale (wird auch als Composite Video bezeichnet).

## 7.18 Belegung des Steckverbinders PF6 für RESET, NMI und HALT

| PIN Nr. | belegt mit |
|---|---|
| 1 | Masse |
| 2 | HALT-LED |
| 3 | Masse |
| 4 | RESET |
| 5 | frei |
| 6 | +5V |
| 7 | Masse |
| 8 | NMI |
| 9 | Masse |
| 10 | frei |

# 8 Sondertypen des SAM-68K

## 8.1 Einleitung

Für spezielle Anwendungen kann die SAM-68K-Platine auch mit einer Taktfrequenz von 10 MHz und/oder einer erhöhten Grafikauflösung geliefert werden. Versionen mit 10 MHz Takt und normaler Grafikauflösung sind jedoch nicht realisierbar.

Die Unterschiede zu den Normaltypen beschränken sich im wesentlichen auf die zwei Punkte "Prozessortakt" und "Grafikauflösung", was jedoch z.T. weitreichende Auswirkungen hat. Diese Unterschiede werden im folgenden besprochen. In allen anderen Punkten gilt die Beschreibung in den Kapiteln 1 bis 7 unverändert.

## 8.2 Auswirkungen des 10 MHz Prozessortaktes

Durch den gegenüber der Normalversion erhöhten Prozessortakt ergibt sich ein Geschwindigkeitsvorteil bei Speicherzugriffen im normalen RAM (RAM-Block 1) und im Video-RAM (RAM-Block 2). Dies sorgt neben der natürlich schnelleren prozessorinternen Verarbeitung von Befehlen für eine insgesamt erhöhte Rechnerleistung.

Bei Erweiterung des Systems mit externen Karten ist zu beachten, daß diese für 10 MHz Systemtakt ausgelegt sein müssen!

Der erhöhte Prozessortakt bedingt gleichzeitig auch eine Erhöhung des E-Taktes von 0.8 MHz auf 1.0 MHz. Dadurch verändern sich A L L E Zeitkonstanten im System, die durch Timer in Peripheriebausteinen gebildet werden.

## 8.2 Organisation des VIDEO-RAMs bei erhöhter Videoauflösung

kws unterstützt eine Videoauflösung von 768 x 480 Punkten im interlaced Mode. Diese Auflösung ist bei Systemen mit 10 MHz Prozessortakt zwingend, bei Systemen mit 8 MHz Prozessortakt optional möglich. Durch den Betrieb im Interlaced Mode ist ein Monitor mit langnachleuchtendem (P39) Phosphor unumgänglich.

Die Zuordnung von Adresse im Videospeicher zu Position auf dem Bildschirm ist nicht mehr - wie bei den normalen Aufloesungen - linear, sondern etwas komplizierter:

1. Bytes werden als acht horizontal benachbarte Pixel abgebildet. Bit 7 ist links im Bild. Ein Bit auf log. 1 ergibt einen hellen Punkt.

2. Das erste Byte der Videoseite erscheint links oben im Bild, das zweite Byte rechts daneben.

3. Eine Zeile des Bildes wird im folgenden Rasterzeile genannt. Die 480 Rasterzeilen werden in 30 Gruppen zu je 16 Rasterzeilen unterteilt. Diese Gruppen werden im folgenden als Balken bezeichnet. Jeder Balken benoetigt 768/8*16 = 1536 (hexadezimal $600) Bytes Speicherplatz.

4. Die einzelnen Balken des Bildes sind von oben nach unten im Videospeicher mit aufsteigenden Adressen abgelegt. Die Anfangsadresse eines Balkens MUSS (!) in den letzten acht Bits mit $00 enden. Danach ergibt sich fuer die Balken folgende Speicherzuordnung:

| Balkennummer | Anfangsadresse |
|---|---|
| 0 | $34C00 |
| 1 | $35200 |
| 2 | $35800 |
| usw. bis | |
| 28 | $3F400 |
| 29 | $3FA00 |

5. Adresszuordnung innerhalb eines Balkens:

Die folgenden Adressen sind hexadezimal als Offset zur Anfangsadresse des Balkens angegeben.

| Nummer der Rasterzeile | horizontale Position der Bytes 0 | 31 | 32 | 63 | 64 | 95 |
|---|---|---|---|---|---|---|
| 0 | $000 | $01F | $020 | $03F | $200 | $21F |
| 1 | $040 | $05F | $060 | $07F | $240 | $25F |
| 2 | $080 | $09F | $0A0 | $0BF | $280 | $29F |
| 3 | $0C0 | $0DF | $0E0 | $0FF | $2C0 | $2DF |
| 4 | $100 | $11F | $120 | $13F | $300 | $31F |
| 5 | $140 | $15F | $160 | $17F | $340 | $35F |
| 6 | $180 | $19F | $1A0 | $1BF | $380 | $39F |
| 7 | $1C0 | $1DF | $1E0 | $1FF | $3C0 | $3DF |
| 8 | $220 | $23F | $400 | $41F | $420 | $43F |
| 9 | $260 | $27F | $440 | $45F | $460 | $47F |
| 10 | $2A0 | $2BF | $480 | $49F | $4A0 | $4BF |
| 11 | $2E0 | $2FF | $4C0 | $4DF | $4E0 | $4FF |
| 12 | $320 | $33F | $500 | $51F | $520 | $53F |
| 13 | $360 | $37F | $540 | $55F | $560 | $57F |
| 14 | $3A0 | $3BF | $580 | $59F | $5A0 | $5BF |
| 15 | $3E0 | $3FF | $5C0 | $5DF | $5E0 | $5FF |

Dazu ein Beispiel:

Gesucht sei die Adresse des Pixels in der (von oben gerechnet)

185. Rasterzeile mit der (von links gerechnet) horizontalen Position als 290. Pixel.

Die 185. Rasterzeile ist die 9. Zeile im 11. Balken. Der 11. Balken hat die Startadresse $34C00 + 11*$600 = $38E00.

Das 290. Pixel ist das Bit 5 des 36. Bytes der Rasterzeile. (Die Zaehlung der Pixel beginnt ebenfalls bei 0.) Das 36. Byte der 9. Rasterzeile hat die Adressdistanz $444 zur Startadresse des Balkens.

Damit ergibt sich die Gesamtadresse des Pixels zu: Bit 5 des Bytes mit der Adresse $39244 (= $38E00 + $444).

## 8.4 Monitorprogramm für die Sondertypen

Die geaenderte Videoauflösung macht den Einsatz eines speziellen Monitorprogramms erforderlich. Es ist erkennlich am Zusatz "/10" in seiner Meldung. Alle seine Funktionen sind identisch mit denen der SAM-68K Normaltypen.

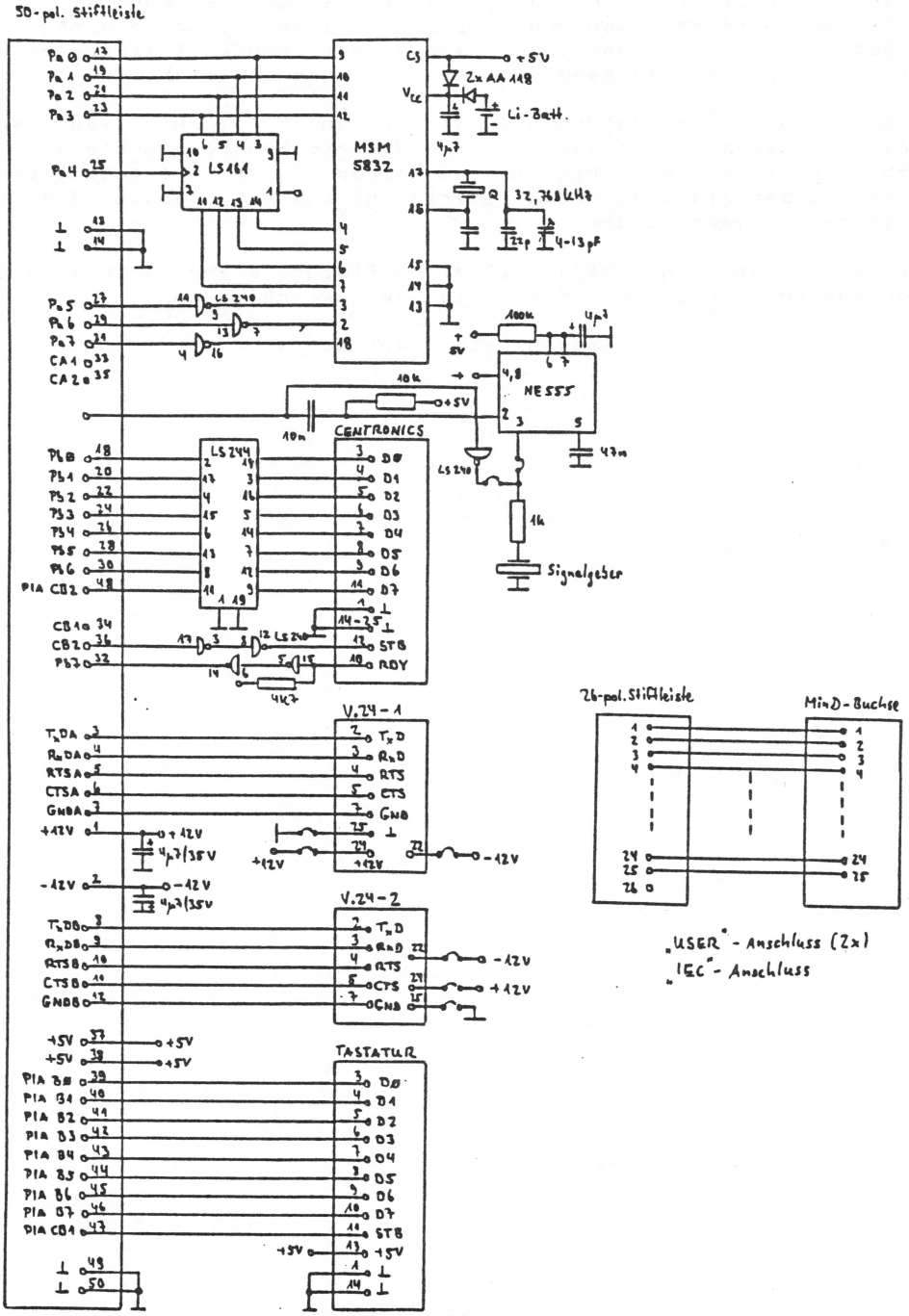
Ausnahme:
Die Videotreiber besitzen ein Charactergenerator im 8 x 16 Raster. Dadurch lassen sich 30 Zeilen mit je 96 Buchstaben auf einer Bildschirmseite abbilden.

Der Beginn des auf den Bildschirm abgebildeten Speicherbereichs ist in Kap. 7.14 angegeben.

Ein Listing dieses speziellen Monitorprogramms ist im Lieferumfang enthalten.

# 9 Kabelplatine (Rueckwandverdrahtung RTC 1)

## 9.1 Schaltplan der RTC 1, Version 4



## 9.2 Anmerkungen zum Schaltplan

Die Centronics-Schnittstelle kann mit 8 Datenbits betrieben werden, dann dient CB2 der PIA 6821 als achtes Datenbit. Dieses achte Bit wird aber von den Monitorprogrammen nicht unterstuetzt. Im Betrieb mit SUSY laesst sich ein entsprechendes Treiberprogramm einbinden, das die Ausgabe mit 8 Datenbits ermoeglicht.

Vorsicht bei der Verwendung der freien 25 pol. Min D Buchsen ! Diese Buchsen sind mit den zugehoerigen Pfostensteckverbindern so verbunden, dass die Pinnummer erhalten bleibt. Die geometrische Anordnung der Pins und deren Numerierung ist bei diesen Stecker-Typen jedoch verschieden.

Die Rückwandplatine wird mit einem Flachbandkabel 1 zu 1 an den Pfostenstecker PF2 der SAM-68K-Platine angeschlossen.

/--- Ende SAM-68K Handbuch ---/

# TASTATUR
# für SAM 68k

(C) Copyright 1982 by
kws Computersysteme GmbH
Rheinstrasse 104
D 7505 Ettlingen

W.Germany

Die in diesem Handbuch zusammengestellten Informationen wurden sorgfältig geprüft und sollten fehlerfrei sein. Dennoch übernimmt KWS keine Verantwortung für Schäden oder Schwierigkeiten, die sich daraus aufgrund von Ungenauigkeiten oder Irrtümern bei der Anwendung oder dem Gebrauch der Produkte ergeben. KWS behält sich das Recht vor, Änderungen an Hard- und Software durchzuführen, die die Zuverlässigkeit und Funktionsweise verbessern oder die Bedienung erleichtern. Diese Änderungen müssen nicht in jedem Einzelfall dokumentiert werden. Alle Produkte von KWS unterliegen dem Urheberrecht oder sind gesetzlich geschützt. Sie dürfen nicht ohne Lizenzvertrag kopiert oder nachgebaut werden.

## Allgemeine Beschreibung

kws bietet für SAM 68K eine Tastatur an, die höchste Ansprüche hinsichtlich Ergonomie, Zuverlässigkeit und Modularität erfüllt.

**Technische Daten:**

- sehr flache Bauweise (30 mm) für ermüdungsfreie Bedienung
- hochwertige, kontaktlose Tasten (Hall-Effekt) mit hoher Lebensdauer ( 10**8 Schaltspiele). Wartungs- und prellfrei.
- Alfanumerisches Tastenfeld nach DIN 2137
- Beschriftung für internationalen Zeichensatz
- Möglichkeit zur Umbelegung auf deutsche Norm (ä,ü,ö usw.)
- separater Zehnerblock
- separater Cursorblock mit Editiertasten
- 20 zusätzliche Funktionstasten, unter SUSY-Betriebssystem frei programmierbar
- N-Key-Rollover
- Autorepeat
- Caps-Lock- und Shift-Lock-Funktion
- 3 Kodierebenen
- eigener Mikroprozessor mit EPROM zur Kodierung
- Modulare Bauweise: Tastatur kann auch nachträglich ohne Gehäuseänderung mit zusätzlichen Tasten erweitert werden
- Auswechselbare Tastenkappen, Sonderbeschriftung möglich
- nur +5V Versorgung nötig
- paralleler Datenausgang (8 Bit) mit Strobe

## Belegung des 25pol. Tastatursteckers

| CANNON Stecker Pin Nr. | Funktion |
|---|---|
| 1 + 14 | Masse |
| 3 | Bit 0 |
| 4 | Bit 1 |
| 5 | Bit 2 |
| 6 | Bit 3 |
| 7 | Bit 4 |
| 8 | Bit 5 |
| 9 | Bit 6 |
| 10 | Bit 7 |
| 11 | Strobe |
| 13 | + 5V |

alle anderen Leitungen reserviert

## Beschreibung der Tastenbelegung

| Taste Nr. | Funktion |
|---|---|
| 1 bis 10 | frei benutzbare Funktionstasten |
| 11 | Umschaltung auf ASCII-Zeichensatz |
| 12 | Umschaltung auf deutschen Zeichensatz |
| 13 | reserviert |
| 14 bis 16 | reserviert |
| 17 | Delete Character |
| 18 | Delete Rest of Line |
| 19 | Insert Line |
| 20 | Cursor Home Up |
| 21 | Character Insert Mode off |
| 22 | Cursor Home Down |
| 23 | Character Insert Mode on |
| 24 | Dezimalpunkt |
| 25 | Umschaltung Command-/Edit-Mode |

Die Tasten 17 bis 23 und 25 sind nur im Zusammenhang mit dem SCREEN-EDITOR unter dem SUSY-Betriebssystem wirksam. Andere Programme können diese Tasten mit anderen Funktionen belegen. Je nach Version der Tastaturtreiber oder Screen-Editor Steuertabelle können sich ebenfalls andere Zuordnungen ergeben.

Hinweis:

Einige von der Tastatur gesendete Zeichen werden von den zugehörigen Tastaturtreibern in andere Codes umgewandelt. Siehe hierzu jeweils die entsprechenden Listings.

**Programmierung der Funktionstasten**

Die Funktionstasten der Tastatur (oberste Tastenreihe) sind unter dem SUSY-Betriebssystem programmierbar. Das bedeutet, daß bei Betätigung einer dieser Tasten bestimmte vordefinierbare Zeichen oder Zeichenfolgen an das Betriebssystem gesendet werden. So kann beispielsweise die Funktionstaste F1 mit der Zeichenfolge "DIR 1" belegt werden; wenn dann auf das Betriebssystemprompt ">" hin diese Taste gedrückt wird, so wird automatisch das Directory des Drives 1 ausgegeben. Auf diese Weise ist es also zum einen möglich, verschiedene Kommandos, die häufig vorkommen, mit einem einzigen Tastendruck aufzurufen, zum anderen können auch einzelne spezielle Steuerzeichen (z.B. Controlzeichen oder ESCAPE - Sequenzen) an Anwenderprogramme weitergereicht werden, wenn diese es benötigen.

Beachten Sie jedoch, daß diese Programmierbarkeit keine Eigenschaft der Tastatur ist, sondern durch das SUSY Betriebssystem nachgebildet wird!

**Wie werden die Funktionstasten programmiert?**

Hierzu gibt es zwei Möglichkeiten:

- "manuelle" Programmierung mit interaktiver Benutzerführung
- Automatische Programmierung mit Hilfe eines Files

**1. Manuelle Programmierung**

Hierzu dient das Betriebssystemkommando

@@SET FK ("setze Function-Keys")

Die Programmierung erfolgt daraufhin im Dialog, wobei folgende Möglichkeiten bestehen:

a) Der Taste wird eine Zeichenfolge zugewiesen, die beim Senden mit <cr> ($0D) abgeschlossen werden soll:

Die entsprechende Zeichenfolge wird nach Betätigung der gewünschten Funktionstaste eingegeben und die Eingabe mit <cr> abgeschlossen.

b) Der Taste wird eine Zeichenfolge zugeordnet, aber es soll kein abschließendes <cr> gesendet werden:

Wie a), als letztes Zeichen der Zeile wird jedoch ein "↑" eingegeben.

c) Es sollen auch "Sonderzeichen" gesendet werden, z.B. auf eine Funktionstaste mehrere durch <cr> getrennte Einzelkommandos gelegt werden:

Um ein Sonderzeichen zu senden, wird ein "↑" vor das entsprechende (Zeichen+$40) gestellt. Man schreibt also: ↑M für $0D usw.

Soll eine Funktionstaste, die zuvor belegt war, wieder völlig desaktiviert werden, so gebe man als erstes und einziges Zeichen ein "↑" ein.

**ACHTUNG**

Das @@SET FK Kommando darf nur angewendet werden, wenn sich das Betriebssystem selbst mit seinem '>' Prompt gemeldet hat und NICHT z.B. wenn man sich im EDITOR befindet! Es vergößert nämlich den den residenten Teil des Betriebssystems und kann dabei im Speicher liegende Programme überschreiben. Also unbedingt Vorsicht walten lassen!

## 2. Automatische Programmierung

Wenn das Betriebssystem gebootet wird, sucht es auf Drive 0 nach einem File mit dem Namen FKEYFILE.SYS. Dieses File muß die Extension .SYS tragen, ist aber ansonsten ein ganz normales, mit einem Editor erstelltes Textfile. Findet das Betriebssystem ein solches File, so wird die Zeichenfolge, die in dessen 1.Zeile steht, der Funktionstaste 1 zugewiesen. Ebenso wird der Inhalt der n-ten Zeile der Funktionstaste n zugewiesen usw. Ist die zu einer Taste gehörende Zeile leer, so wird diese Funktionstaste nicht programmiert.

Neben dieser Möglichkeit, die Funktionstasten waehrend des Bootens des Betriebssystems zu programmieren, gibt es noch eine zweite Möglichkeit.

Mit dem Befehl

@@ INS filename.TXT /FK

können die Funktionstasten mit Hilfe eines vorbereiteten Files auch neu- oder umprogrammiert werden, ohne daß das Betriebssystem neu gebootet und ein neues FKEYFILE.SYS erzeugt werden muß.

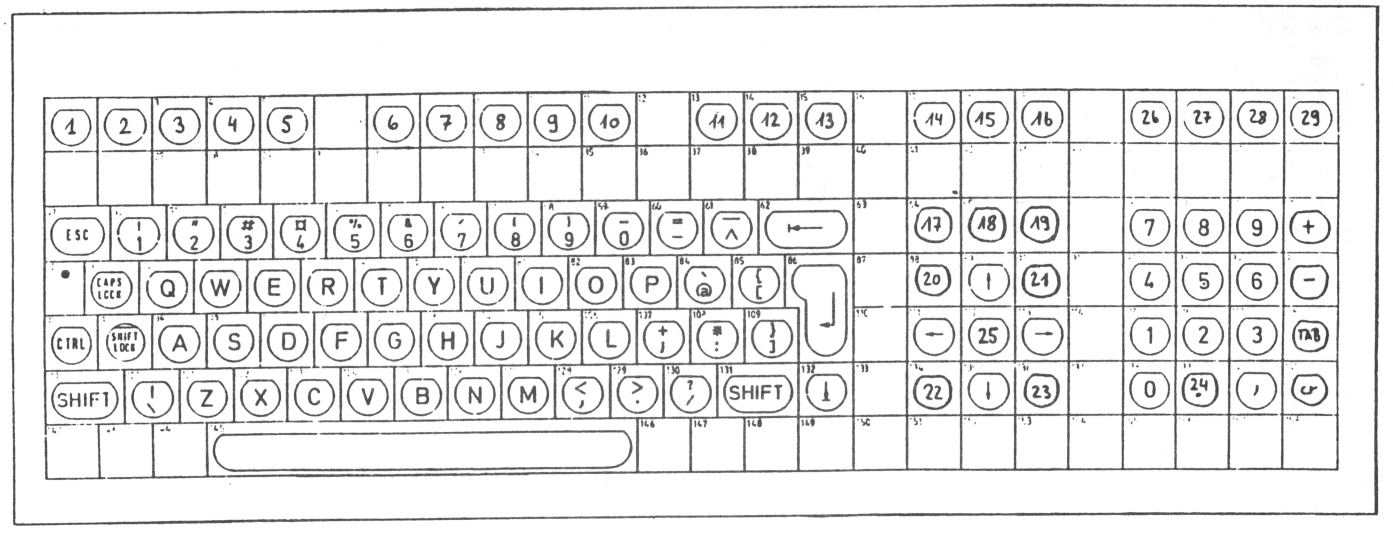
**ACHTUNG**

Dieses Kommando darf ebenfalls nur dann angewendet werden, wenn

sich das System selbst mit seinem '>' Prompt gemeldet hat und NICHT z.B. wenn man sich im EDITOR befindet! Es vergößert nämlich den den residenten Teil des Betriebssystems und kann dabei im Speicher liegende Programme überschreiben. Also unbedingt Vorsicht walten lassen!

**ANLAGEN**

Anordung der Tasten
Code-Belegung der Tasten
Schaltplan
Optionen




Taste

Abdeckung

Leuchttaste

Abdeckung für LED

| | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 8A | 8B | 8C | 8D | 8E | 8F | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 |
| 98 | 99 | 9A | 9B | 9C | 9D | 9E | 9F | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | AA | AB | AC | AD | AE | AF |
| 1B | 21 31 | 22 32 | 23 33 | 24 34 | 25 35 | 26 36 | 27 37 | 28 38 | 29 39 | 5F 30 | 3D 2D | 7E 5E | 08 | | 80 | 81 | 82 | 83 | 84 | 37 | 38 | 39 | 85 |
| 0 | CAPS LOCK | 11 51 71 | 17 57 77 | 05 45 65 | 12 52 72 | 14 54 74 | 19 59 79 | 15 55 75 | 09 49 69 | 0F 4F 6F | 10 50 70 | 60 40 | 7B 5B | 0D | 86 | 87 | 08 | 88 | 89 | 34 | 35 | 36 | 8A |
| CTRL | SHIFT LOCK | 01 41 61 | 13 53 73 | 04 44 64 | 06 46 65 | 07 47 67 | 08 48 68 | 0A 4A 6A | 0B 4B 6B | 0C 4C 6C | 2B 3B | 2A 3A | 7D 5D | | 8B | 08 | 0C | 09 | 8C | 31 | 32 | 33 | 8D |
| SHIFT | 7C 5C | 1A 5A 7A | 18 58 78 | 03 43 63 | 16 56 76 | 02 42 62 | 0E 4E 6E | 0D 4D 6D | 3C 2C | 3E 2E | 3F 2F | SHIFT | | 0A | 8E | 8F | 0A | C0 | C1 | 30 | C2 | 2C | C3 |
| C4 | C5 | C6 | 20 | | | | | | | | C7 | C8 | C9 | CA | CB | CC | CD | CE | CF | D0 | D1 | D2 | D3 |

Tastenpos

CTRL - Code

Taste ist durch CAPS - LOCK beeinflußt

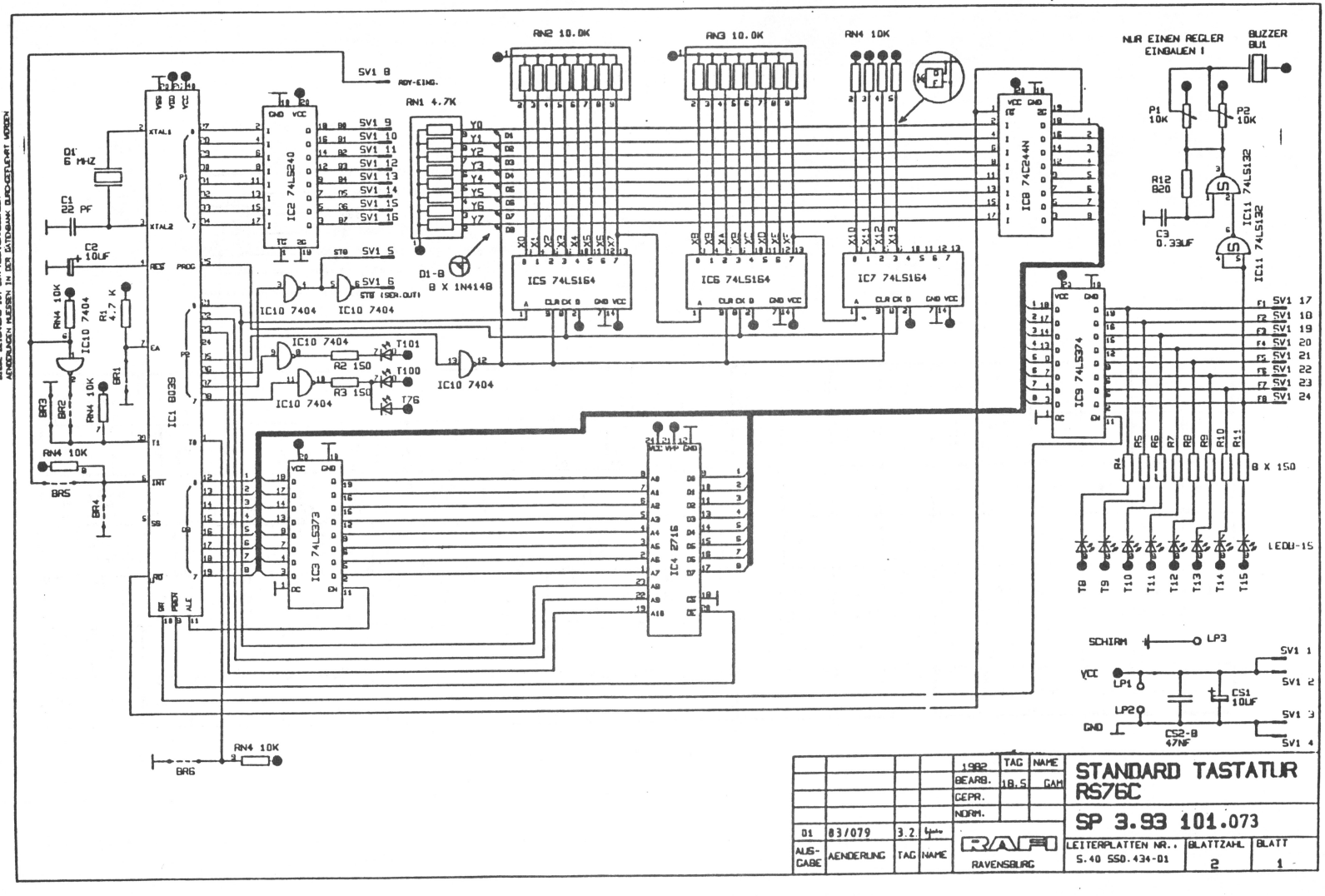
SHIFT - Code

UNSHIFT - Code

Taste ist ohne DS gesperrt

kein Autorepeat

DIESE ZEICHNUNG IST EIN RECHNERWERK
ÄNDERUNGEN MÜSSEN IN DER DATENBANK DURCHGEFÜHRT WERDEN



| | | |
|---|---|---|
| Negative Logik | | Diode 1 |
| Gerade Parität | | Diode 2 |
| Kein Autorepeat | | Diode 3 |
| Codebelegung Deutsch | | Diode 4 |
| Ohne Parität | | Diode 5 |
| Datenausgabe Seriell | 110 Baud | Diode 6 |
| Datenausgabe Seriell | 300 Baud | Diode 7 |
| Datenausgabe Seriell | 600 Baud | Diode 6 + 7 |
| Datenausgabe Seriell | 1200 Baud | Diode 8 |
| Datenausgabe Seriell | 2400 Baud | Diode 6 + 8 |
| Datenausgabe Seriell | 4800 Baud | Diode 7 + 8 |
| Datenausgabe Seriell | 9600 Baud | Diode 6 + 7 + 8 |